FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Gunjan Chandra

IMPACTS OF DATA SYNTHESIS: A METRIC FOR QUANTIFIABLE DATA STANDARDS AND PERFORMANCES

Master's Thesis
Degree Programme in Computer Science and Engineering
June 2020

Chandra G. (2020) Impacts of Data Synthesis: A Metric for Quantifiable Data Standards and Performances. University of Oulu, Degree Programme in Computer Science and Engineering, 67 p.

ABSTRACT

Publicly shared data could unfold a wide range of innovative pedagogical and learning techniques. In the case of healthcare, open data could save lives. Consolidating medical data with lifestyle information can support possibilities for further development of current approaches towards medical diagnoses and treatments. It is critical to note that healthcare data contains sensitive information about patients and therefore, could lead to harmful consequences if such details reach the wrong hands. The use of the concept of data anonymisation for reducing the risk of disclosure to share data publicly is the standard practice. However, current data anonymisation techniques have failed multiple times in the past. The goal of this study is to evaluate the performance of an emerging practice for data sharing, by utilising a tool for data synthesis, termed *Synthpop*. The synthetic data is generated by executing the multiple imputation methods, although differently. This study describes and analyses *Synthpop* by establishing the data standards and measuring the impacts of the data synthesis process based on the utilities and quality of information contained in the data. The analyses reveal that synthetic data simulates original data by adequately preserving the utilities and quality of the information content.

Keywords: Synthpop, Data Sharing, Privacy, Data Anonymisation, Machine Learning, Data Utility, Entropy, Information, Data Quality, Type 1 Diabetes, Human Activity Recognition.

TABLE OF CONTENTS

ABSTRACT
TABLE OF CONTENTS
FOREWORD
LIST OF ABBREVIATIONS AND SYMBOLS
1. INTRODUCTION 7
2. RELATED WORK 9
2.1. Benefits of Open Healthcare Data Sets 9
2.2. Importance of the Subject's Privacy 10
2.3. Data Anonymisation 10
3. METHODOLOGY 14
3.1. Synthpop 14
3.1.1. Basic Functionality 14
3.1.2. Methods for Synthesis 15
3.1.3. Controlling the Sequence and Prediction 17
3.1.4. Handling Data with Restricted or Missing Values 18
3.2. Utility Measures of Data 18
3.2.1. General Utility Measures 19
3.2.2. Specific Utility Measures 20
3.3. Quality of Information Content 25
3.3.1. Entropy 25
3.3.2. Mutual Information 26
3.4. Data Sets 27
3.4.1. DIPP Data Set 27
3.4.2. HAR Using Smartphone Data Set 28
4. EXPERIMENTS AND RESULTS 33
4.1. DIPP Data Set 33
4.1.1. Specific Utility 34
4.1.2. General Utility 39
4.1.3. Quality of Information Content 43
4.1.4. Outlines 44
4.2. HAR Using Smartphone Data Set 45
4.2.1. Specific Utility 45
4.2.2. General Utility 52
4.2.3. Quality of Information Content 54
4.2.4. Outlines 55
5. DISCUSSION 57
5.1. Principal Discoveries 57
5.1.1. DIPP Data Set 57
5.1.2. HAR Data Set 58
5.2. Synopsis and Future Work 60
6. CONCLUSION 61
7. REFERENCES 62

FOREWORD

This study was conducted under and supported by the Biomimetics and Intelligent Systems Group (BISG) of the Computer Science and Engineering Department of the Faculty of Information Technology and Electrical Engineering at the University of Oulu, Oulu, Finland.

I want to express my most profound appreciation to my thesis supervisor Dr Pekka Siirtola and reviewer Dr Satu Tamminen, for their constructive feedback, patience, and valued guidance. Special thanks to Dr Eija Ferreira for her insightful suggestions regarding the growth and completion of the study. I would further like to extend my gratitude to Dr Ian Oliver and his team for presenting me with the grounds concerning the Mutual Information analyses. Additionally, I am indebted to MSc Oana M. Stoicescu for providing me with the assistance and means to complete this work. Finally, appreciation to Johan Estiévenart for supporting me throughout the process.

Oulu, June 16th, 2020

Gunjan Chandra

LIST OF ABBREVIATIONS AND SYMBOLS

Abbreviations

ADL	Activities of Daily Living
ALE	Accumulated Local Effects
ANN	Artificial Neural Networks
CART	Classification And Regression Tree
DIPP	Type 1 Diabetes Prediction and Prevention
EHR	Electronic Health Record
FFT	Fast Fourier Transform
FI	Feature Importance
FN	False Negative
FP	False Positive
GADA	Glutamic acid decarboxylase
GBM	Gradient Boosted Machine
GDPR	General Data Protection Regulation
HAR	Human Activity Recognition
HCC	Human-Centered Computing
HIPAA	Health Insurance Portability and Accountability Act
IA2A	Protein tyrosine phosphate autoantibody
IAA	Antibodies of insulin
IBM	International Business Machines
IDC	International Data Corporation
k-NN	k-Nearest Neighbour
LDA	Linear Discriminant Analysis
MI	Mutual Information
NN	Neural Networks
PPMCC	Pearson product-moment correlation coefficient
SYLLS	Synthetic Data Estimation for UK Longitudinal Studies
T1D	Type 1 Diabetes
TN	True Negative
TP	True Positive
t-SNE	T-distributed Stochastic Neighbor Embedding
UMAP	Uniform Manifold Approximation and Projection
ZB	Zettabytes

Symbols

C^*	Comparison function
$C(y\|k)$	Cost of classifying an observation as y when its true class is k
D	Original Data set
D^*	Kolmogorov-Smirnov statistic
$\hat{F}$	empirical distribution
G	Tangent sigmoid transfer function
H_o	Null hypothesis
H_a	Alternative hypothesis

$H(X)$	Entropy of X
$H(X\|Y)$	Entropy of X given Y
$I(X;Y)$	Mutual information between X and Y
L	Loss function
M	Number of iterations
P, p	Probability of occurrence
$P(k)$	Prior probability of class *k*
$\hat{P}(k\|x)$	Posterior probabilty of class *k* for observation x
R	Correlation coefficients
S_i	Synthetic data sets
$W^{(l)}$	Weights matrices
X, Y	Random variables
$argmin$	Argument of the maximum
$b^{(l)}$	Bias vectors
$dist$	Distance
exp	Exponential
i	Target class vector
j	Weak learner
k	Positive integer
log	Logarithmic
n	Matrix of net input (column) vectors
s	Soft max transfer function
sum	Summation
sup	Supremum function
t	Vector of tests
x	Matrix (n x p) of original covariates
xp	Matrix (k x p) of synthesised covariates
y	Original data vector of length n
$\hat{y}$	Predicted classification
$\sum$	Summation
α	Threshold value for level of significance
γ_{im}	Residuals
ε	Privacy loss parameter
ρ	Population correlation coefficient
σ	Standard deviation
Ψ	Di-gamma function

1. INTRODUCTION

In today's digital era, with the exponential growth in the quantity and quality of data collection methods, over 2.5 Quintilian bytes of data are created every single day [1], and International Data Corporation (IDC) predicts that it is only going to grow from there, from 33 Zettabytes (ZB) in 2018 to 175 ZB by 2025 [2]. By 2020, there will be 44 ZB of data, suggesting 40 times more bytes of data than there are stars in the observable universe [3]. Regardless of the size, data helps us in solving problems, making better decisions, maintaining performances, and improving existing processes [4]. Despite the benefits, as of 2013, only 0.5% of the total data was analysed [5].

To keep up with the speed of data generation, data collectors could make the data publicly available and target more analysts around the globe. Open data sets have many other advantages as well, ranging from consolidating different data sets for finding new knowledge to verify previously made findings. Currently, to fulfil these requirements, researchers duplicate the data collection, which is an unnecessary use of resources. Private data also restrict scholars to share in-depth knowledge of the topic and impose a limitation on communication. Transparency in the research community is not only going to help in the advancement of the technology but will also facilitate better opportunities for innovations and solving current problems.

However, making the data set publicly available increases the risk of disclosure. In the survey done by Morey T., Forbath T., and Schoop A., more than 72% of United States citizens reported being worried about sharing personal information online [6]. If data reaches the wrong hands, sensitive information can be exploited for blackmailing, mass surveillance, social engineering, or identity theft [7]. All these risks obtrude data collectors and researchers from sharing the data and opt for proprietary data policy. Not alone researchers but also students suffer from this; for example, teaching data analysis with medical data such as Electronic Healthcare Record (EHR) is significantly restrained by laws protecting the patients' privacy in many countries [8]. Notwithstanding the benefits, the limitations imposed by these laws hinder innovation and limit educational opportunities.

Sensitive data as defined by General Data Protection Regulation (GDPR) can be any data that reveals the racial or ethnic origin of a person, political opinions, religious or philosophical beliefs, trade union membership, genetic data, or use biometric data to uniquely identify a person and data concerning health or a person's sex life or sexual orientation [9]. After GDPR came into force on May 25th 2018, the collection, sharing, and processing of data has become more secure than ever but, at the same time, data acquisition has become more difficult for researchers. Determining whether a subjects' consent is required for secondary data use in research, and which forms to be filled, further slows down the process.

In the healthcare sector, medical data either collected as part of clinical research or recorded during clinical practice is considered as confidential and needs to be pseudonymised or anonymised before leaving the hospital [10]. Traditional pseudonymisation and anonymisation techniques consist of the removal of identities such as names, addresses, and national identity numbers. This method solves the issue of direct identification, but a person can still be re-identified when data is re-linked to other data sets, and the risk in the reduction of k-anonymity can be expected [11, 12]. To further reduce the risk of re-identification, data scientists use data aggregation

techniques and induce random noise to the data, which leads to distortion of the relationship between variables in the data set [11]. Having a data set with distorted relationships between variables can be misleading [13]. As the correlation between the features changes, the risk that correlation is interpreted as causation increases and can lead to misconceptions [14].

Another problem could come from a perspective of different expertise as one researcher has the data but does not have machine learning knowledge, for example. Not being able to share data, in this case, could hinder data exploration. The generation of the synthetic data set, which preserves the statistical properties of the original data set and, simultaneously, ensures the patient's privacy, will be the fittest case in the current scenario to share data. In this study, a data synthesis tool, an R package termed **Synthpop** will be explored and examined while underlining the statistical properties, machine learning applicability, and quality of information contained in the data set. The primary objective will be to question the performance of the synthesis tool by evaluating the impact of data synthesis procedure; Over two different data sets for comprehensiveness evaluation. The first data set is the Finnish Type 1 Diabetes Prediction and Prevention (DIPP) study database [15], pre-processed by M.Sc. Oana Maria Stoicescu and second is the Human Activity Recognition (HAR) data set from the University of California Irvine machine learning repository [16]. Impacts of data synthesis will be measured based on the general and specific utility, and quality of information of the synthetic data set compared to the original data set. General utility measures will evaluate the difference in the statistical properties of the data sets, and specific utility measure will focus on the performance of the fitted models over different data sets (synthetic and original). One null and one alternative hypothesis will be defined, evaluating the difference in the results of utility measures. The **Synthpop** will succeed in a performed test if results fail to reject the null hypothesis, which states that the two data sets (synthetic and original) have at most a statistically non-significant difference. Moreover, the study will be finalised via evaluating from an information-theoretic point of view, by analysing entropy and mutual information within the data sets or in comparison to measure the quality of information contained in the data set.

2. RELATED WORK

2.1. Benefits of Open Healthcare Data Sets

The maintenance of data obtained as a part of the clinical analysis has evolved. Initially, medical data was generated and maintained by health care professionals in the form of several EHR. Nowadays, most countries possess a centralised EHR system to accommodate the availability and completeness of the data. The purpose of centralised EHR is to hold detailed information about the patient's medical archives in one place. These centralised EHR can later be combined with other data sets to help medical professionals administer the best possible treatment with knowledge gained from data by using next-generation technologies. Despite the benefits, few considerable obstacles prevail in the process of exploring and achieving this goal [17]. Some are associated with the content and structure of the modern healthcare database; others regard the complications and expense of producing and sustaining comprehensive databases.

On that account, within the fastest-growing field of science, the collaboration between medical professionals is continuously growing with other doctors, healthcare providers, drug providing companies, and data scientists to study and sustain the data. Trade of knowledge and data is part of the ubiquitous global movement [18]. Furthermore, it is advancing science along with the transformation of healthcare systems and how we make decisions.

In healthcare, open data can save lives by enabling new data-driven technologies, including artificially intelligent systems, and potentially transform medicine. Precision medicine [19] is an emerging model-based technique for patient care that weighs the peculiar variability in genes, environment, and lifestyle to provide treatment tailored to individual characteristics of each patient. Precision medicine involves the amalgamation of both clinical and lifestyle information of patients to provide precise intervention.

An approach such as precision medicine relies on the varieties and sources of data. The performance of the predictive model for acquainting clinical practice depends on the size and diversity of data. Lack of cohort diversity in data can lead to bias and inequity. For example, International Business Machines (IBM) Watson revealed a bias towards treatment provided at the hospital from where originally data was collected [20]. Hence, a practical advantage of open data is not merely that we can use an individual database more extensively; it is the ability to leverage and consolidate with other databases. The study reveals that machine learning models perform far better with more diverse longitudinal data sets [21].

However, those rich veins of data are too often locked away. There are several reasons why every data set is not publicly available. Most often, data collectors do not get the recognition of their investment in data collection; hence the desire to be the first to explore and utilise the data before they sell or distribute it to others. Nevertheless, one being the most important reason is the privacy of the subjects.

2.2. Importance of the Subject's Privacy

After coming across the benefits of open data, mainly governments are spearheading the concept of open databases over the last decade. With open database comes the risk of disclosure and can lead to many harmful consequences. Disclosure risk gets higher with a better privacy attack. Privacy attack is the method of identifying a subject's identity within the data set or by combining multiple databases. Once a subject is identified, the knowledge can be used for blackmailing, mass surveillance, social engineering, or identity theft [7].

Medical history, which includes information about the sexually transmitted disease, substance abuse, psychiatric treatment, or elective abortion, are sensitive pieces of information about a person, and the person may not want to reveal this information to anyone except specialists. A person can also wish to not reveal any private information for no particular reason because they feel invaded and find the entire system distasteful [22].

As per law in most countries, data sharing is possible with the given consent from the data owner, providing that the person's identity will remain anonymous. In order to continue exercising data sharing and collection, many different data anonymisation techniques are brought into play.

2.3. Data Anonymisation

Those who wish to release a version of data publicly opt for traditional data anonymisation techniques. Irreversibly altering the data with the intention of privacy assurance is a method called data anonymisation [10]. Anonymisation can be done by either encrypting or removing personally identifiable information. Data is said to be anonymous if the subject's identity can no longer be identified directly or indirectly.

Data anonymisation approaches have evolved, developed, and adapted to our need multiple times in past. Figure 1 provides a brief history of data anonymisation and the starting point of knowledge discovery. Around 1850, when the US Federal Bureau of Statistics (Census Bureau) started receiving questions about privacy, as a protection measure, the Census Bureau began to remove personal information from publicly available census data. Census Bureau became one of the first to adopt data anonymisation concept by removing explicit identifiers such as name, address, and national identity numbers as this information can be misused, abused, or not comply with the agreement of the data sharing policy mentioned during data collection. Herman Hollerith, famously recognised for founding IBM, invented the tabulating machine for Census Bureau. Later in 1950, Census Bureau became the first to use the tabulating machine to assist in summarising information. In early 1960, as the US government decided to set up the National Data Centre to improve state information system, which the public viewed as contradicting with constitutional rights. The same debate was repeated in Europe—directing the German state of Hesse to introduce the Hessian Data Protection Act in 1970 [23] and the US to pass the Privacy act in 1974 [24].

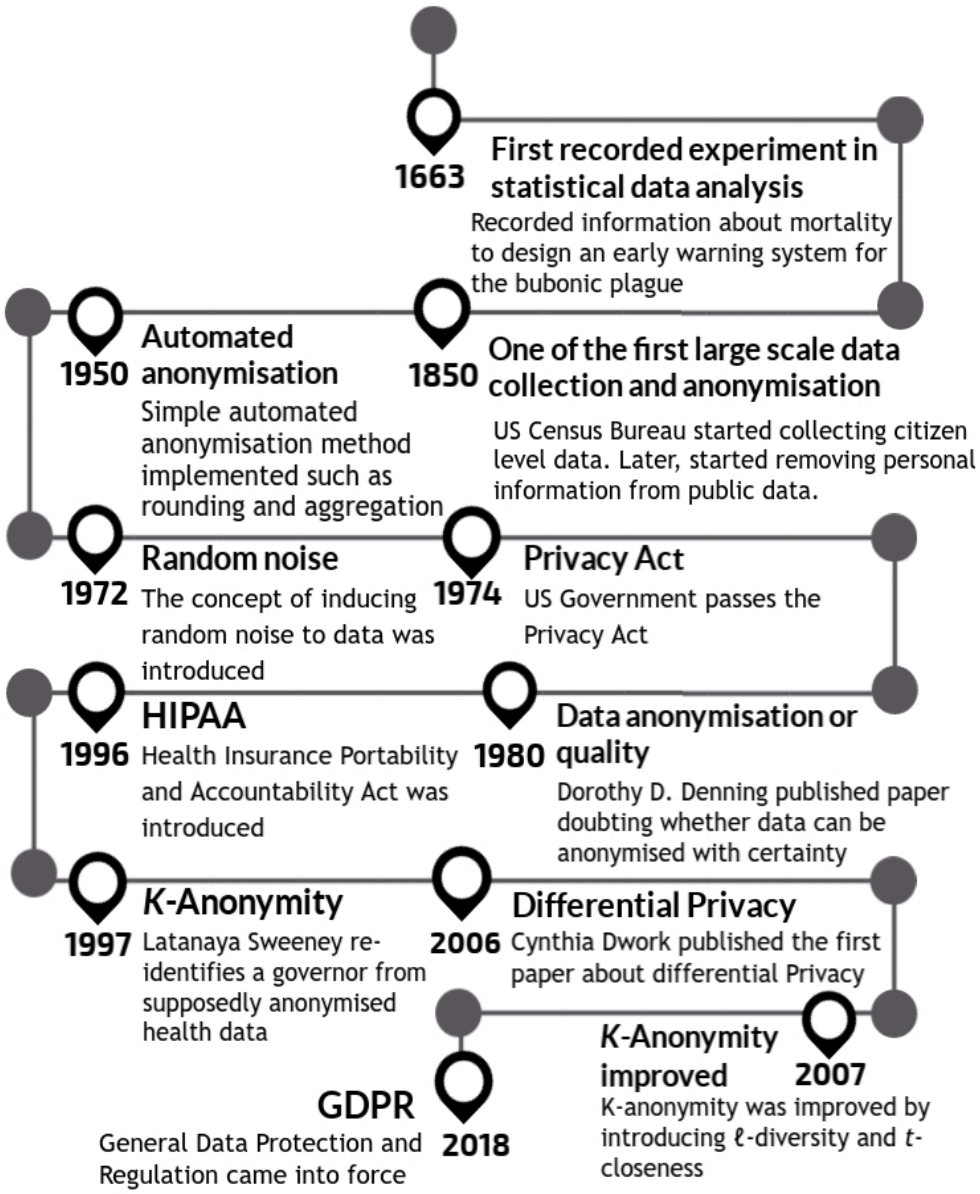


Figure 1. History of data anonymisation

In 1972, a paper proposing the concept of inducing noise to data was published [25]. Later in 1980, researcher Dorothy E. Denning published a paper showing concern whether data can be anonymised with certainty as her analysis showed that "noise" can often be removed by averaging responses for carefully selected query sets [26]. For additional disclosure protection, data aggregation and sampling, alongside with inducing random noise to sensitive variables were standard practices before sharing data. These practices make it challenging to identify an individual, yet the risk of producing data with distorted relationships between variables increases [27].

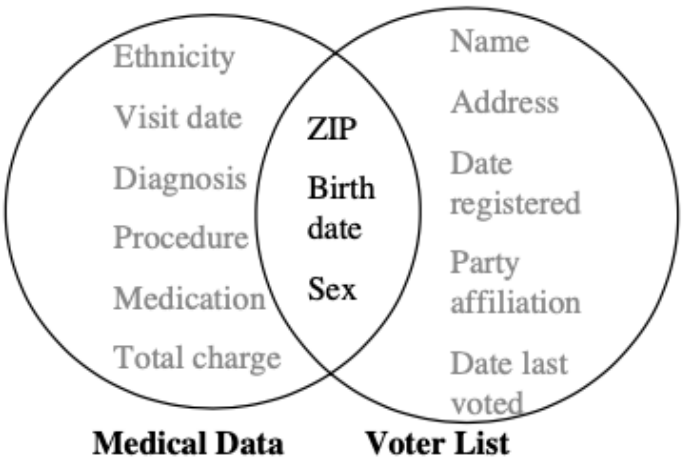


Figure 2. Linking data for re-identification

For almost 15 years until the Health Insurance Portability and Accountability Act (HIPAA) was enacted, the entire computer science community had lost interest in data anonymisation. Soon after in 1997, Latanya Sweeney succeeded to re-identify then Massachusetts Governor from supposedly anonymised health data and presented the concept of *k*-anonymity [28]. Figure 2 represents an example of re-identification through data linking [11]. Later in 2002, L. Sweeney also provided the *k*-anonymity model to overcome the shortcomings of earlier anonymisation techniques [11]. Anonymised data can occupy a quality such as *k*-anonymity [11], and *k*-anonymity can be used as one of the analyses for the level of anonymity. If the information for each individual contained in the anonymous data cannot be distinguished from at least *k-1* individuals whose information also appears in the anonymous data, the data set is considered *k*-anonymity protected [11]—assuming that the anonymised data remains practically useful. Soon after, the *k*-anonymity model was enhanced by introducing ℓ-diversity and *t*-closeness to the model [29, 30].

In 2006, a paper about differential privacy was published stating that privacy can be preserved by calibrating the standard deviation of the noise according to the sensitivity of function f [31]. Differential privacy uses the parameter ε to determine the degree of privacy which is inversely proportional to the value of ε. In other words, for better protection, the value for ε must remain low. After eight years, in 2014, the theory was put into practice by Google as they began to collect differential private user statistics in Chrome [32]. Two years later, Apple started using differential privacy on user data for iPhones [33].

Since a data set with a low ε value can only be queried a few times, questions of utility versus privacy started to emerge. For example, data set with ε value less than one can only be queried around a few 10s of queries in total, after that access to the data is no longer authorised as the privacy can not be assured. Therefore, in order to access the data set more often ε needs to be large, which leads to sacrifice in the data protection. Despite the efforts, there is a growing consensus that traditional anonymisation techniques have proven to fail multiple times in the past [11, 34, 35].

In 2018, The General Data Protection Regulation (GDPR) came into force, allowing the data subjects to decide on the usage and disclosure of their data. Furthermore,

GDPR holds data collectors responsible for evaluating proposed research before sharing the data. This ensures adequate provision to protect the subject's privacy and maintain the confidentiality of the subjects in data set [36]. After understanding the complexity of today's digital databases and how privacy attacks can be personalised and can benefit by consolidating other databases to identify individuals; many researchers, scientist, and mathematicians collaboratively are taking up the task to build and advance data anonymisation procedures to make them suitable for current data needs. A novel tool proposing an alternative approach towards data sharing termed **Synthpop** [37] was utilised later in 2018. A synthesised version of highly sensitive data probing the role of ovulatory changes on sexual desire and behaviour was publicly released [38]. The data set consists of 26 thousand diary entries from women. Considering the sexual diaries are extremely sensitive and hard to anonymise completely, the data collector did not request the consent from participants to make data publicly available, but instead synthesised the data and made it publicly available for secondary data analysis. In this study, the performance of **Synthpop**, that produces a synthetic version of data which is said to be anonymous, will be explored and examined by measuring the impacts of the data synthesis process.

3. METHODOLOGY

Medical data is considered to be highly sensitive and legally owned by patients in many countries. In order for data to be most useful for diagnosis, prognosis, and treatment planning, the identity of the patients must, in most cases, be relinked to the data analytic results. Usually, medical data is anonymised before leaving the hospital, but as the patient's identity is often needed to be relinked, therefore medical data cannot be fully and irreversibly anonymised [10, 12]. Medical data requires complex pseudonymisation procedure to ensure k-anonymity, which is considered legal and is ethically acceptable [11]. Furthermore, especially for healthcare data, more sophisticated tools to measure the impact of data anonymisation are needed. In this chapter, Section 3.1 introduces the primary method of data synthesis, used as a function for data anonymisation and synthesis. Sections 3.2 and 3.3 explain various ways to measure the utilities and the quality of information contained in the generated data, respectively. Moreover, Section 3.4 describes the data sets used in the experiments to analyse the comprehensiveness of the **Synthpop**-generated synthetic data.

3.1. Synthpop

Data synthesis is a process of generating data that mimics the original data set but does not hold any disclosure records. Figure 3 represents the workflow of generating synthetic data in brief and Figure 4 gives details of sub-processes.

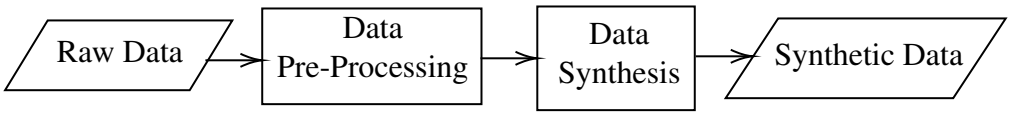


Figure 3. Workflow of generating synthetic data

The tool for data synthesis used in this research is an R package termed **synthpop** [37]. The **synthpop** package was written as a part of the Synthetic Data Estimation for UK Longitudinal Studies (SYLLS) project. Formerly to share the sensitive population-level data outside the setting where researchers were holding the original data set. Later, the **synthpop** package was altered to makes it applicable to other data sets.

3.1.1. Basic Functionality

The method works by replacing some or all observed values by sampling from an appropriate probability distribution, conditional on the variable to be synthesised, the values from all previously synthesised columns of the original data set, and the fitted parameters of the conditional distribution (simple synthesis) or posterior predictive distribution of parameters (proper synthesis) while retaining the statistical properties of the original data set and relationships between the variables. The synthetic data can be produced simply via `syn()` in a single command providing a data set, which is a data frame or matrix to be synthesised. Users can customise the synthesis of a

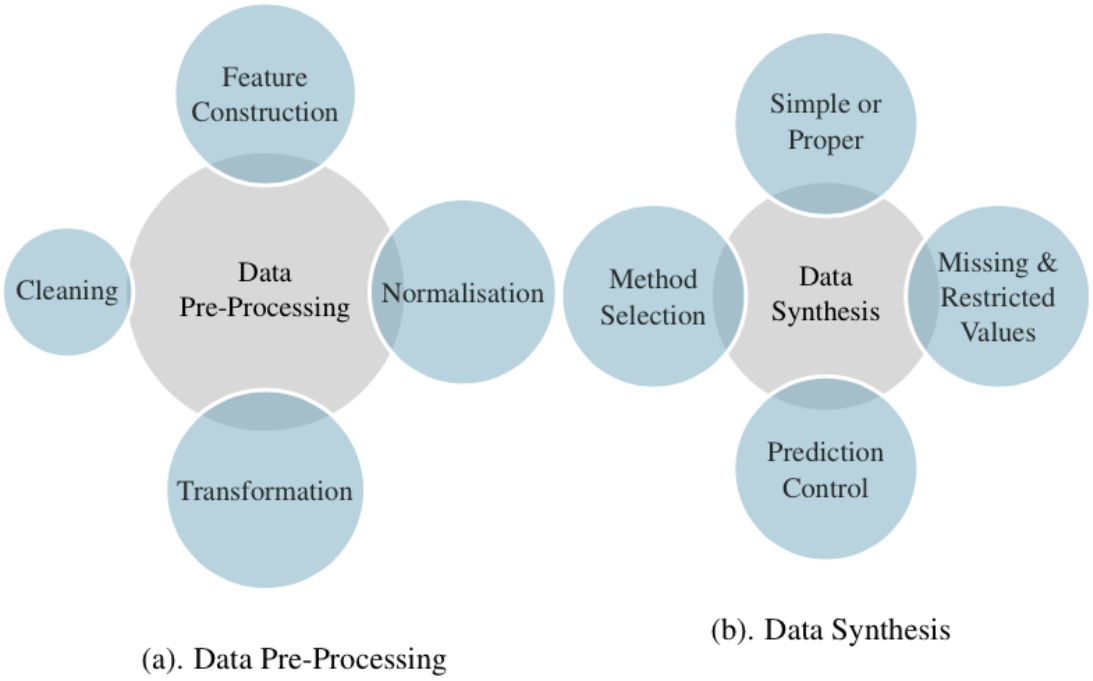


Figure 4. Data pre-processing and synthesis

data set according to requirement, applicability, and type of data variables for better performance of the overall system. By default, the `syn()` function produces one synthetic data set, but multiple data sets can be generated by setting the parameter `m` to a coveted number. An additional parameter `seed` can be used to fix the pseudo-random number generator to reproduce the same results. By default, `syn()` function uses simple synthesis but proper synthesis can be done by setting the `proper` argument to `TRUE`.

3.1.2. Methods for Synthesis

The **Synthpop** consists of both parametric and non-parametric methods. Table 1 lists the methods currently implemented in **Synthpop**. Each method generates synthetic values for each variable sequentially. Synthetic values are generated using the distribution of variable to be synthesised conditional on the distribution of previously observed synthetic and original variables called predictors. The default method of synthesis is `"cart"` for all variables with predictors. The method `"cart"` is a non-parametric method based on Classification And Regression Tree (CART); capable of handling any type of data. However, the first variable to be synthesised in the data set does not have a predictor, and it is a particular case where its values are by default generated by random sampling with replacement from original values (`"sample"` method). However, the user does not need to use the same method of synthesis for all variables with predictors; a user can assign different methods from the list of methods to each variable in the data set befitting the type of data. On the other hand, by setting parameter `method` to `"parametric"` assigns default parametric

methods to each variable based on their data type. Furthermore, if a user does not want to change or synthesise a variable, an empty method (" ") should be used for that variable. Finally, a new method of synthesis can be defined by writing a function named `syn.newmethod()` and for synthesis, specify the `method` parameter of `syn()` as `"newmethod"`.

Table 1. Built-in synthesising methods. * Indicates default parametric methods [37].

Method	Description	Data type
Non-parametric		
`ctree, cart`	Classification and regression trees	Any
`surv.ctree`	Classification and regression trees	Duration
Parametric		
`norm`	Normal linear regression	Numeric
`normrank`*	Normal linear regression preserving the marginal distribution	Numeric
`logreg`*	Logistic regression	Binary
`polyreg`*	Polytomous logistic regression	Factor, >2 levels
`polr`*	Ordered polytomous logistic regression	Ordered factor, >2 levels
`pmm`	Predictive mean matching	Numeric
Other		
`sample`	Random sample from the observed data	Any
`passive`	Function of other synthesised data	Any

Implementation of methods

Let y denote an original data vector of length n, xp denote a matrix (k x p) of synthesised covariates, and x denote a matrix (n x p) of original covariates.

1) Classification tree (`"syn.ctree"`) or **Classification and regression tree** (`"syn.cart"`):

It fits a classification or regression tree by binary recursive partitioning followed by finding a terminal node for each xp. Finally, a donor from the members of the node is randomly drawn and take that draw's observed value as the synthetic value. The difference in `"syn.ctree"` and `"syn.cart"` is that they uses functions from different packages. `"syn.ctree"` uses *ctree* function from **party** package, whereas, `"syn.cart"` uses *rpart* function from **rpart** package. The selection of splitting variables and a stopping rule for the spitting process makes them differ amongst others.

2) Random forest (`"syn.rf"`):

Uses Breiman's random forest algorithm for classification and regression [39]. Furthermore, It utilise *randomForest* function from the **randomForest** package.

3) Bagging (`"syn.bag"`):

Generates synthetic data using bagging by utilising *randomForest* function from the **randomForest** package with number of sampled predictors equal to number of all predictors.

4) Logistic regression (`"syn.logreg"`):

It is used for the synthesis of binary variables by the non-Bayesian or approximate Bayesian logistic regression model. For non-Bayesian method, it first fits a logistic regression to the original data, then calculate the predicted inverse logits for synthesised covariates. Finally, compare the inverse logits to a random (0,1) deviate and obtain synthetic values. For approximate Bayesian method (for proper synthesis), it repeats the same process as for non-Bayesian method with one additional step before computing inverse logits, drawing coefficients from normal distribution with mean and variance estimated in first step.

5) Normal Linear regression preserving the marginal distribution (`"syn.normrank"`):

First synthetic values of Normal deviates of rank of the values in y are generated using the spread around the fitted linear regression line of Normal deviates of rank given x. Then synthetic Normal deviates of ranks are transformed back to get synthetic ranks which are used to assign values from y. Whereas, for proper synthesis, the regression coefficients are drawn from normal distribution with mean and variance from the fitted model.

6) Unordered polytomous regression (`"syn.polyreg"`):

The synthetic categorical variables are generated by the polytomous regression model. First, it fits categorical response as a multinomial model, then it computes predicted categories, and finally, add appropriate noise to predictions. The algorithm uses *multinom* function from **nnet** package. Numerical variables are scaled before fitting to cover the range (0,1).

3.1.3. Controlling the Sequence and Prediction

Synthetic values of each variable are generated from a joint distribution. The joint distribution is defined in terms of a series of conditional distributions. The values are imputed sequentially from the distribution of the variable to be synthesised conditional on two distributions: 1) The distribution of all previously observed variables in the original data set, 2) The distribution of all previously synthesised variables. This sequential process is by default automated, following the order of how variables appear in the data set (left to right). However, the order can be changed or specified for each variable by listing out the indices of columns in the desired order to set parameter `visit.sequence`. If a user wishes not to synthesise a variable and not use it as a predictor, it should be removed from the `visit.sequence`. Furthermore, if a user wishes not to synthesise a variable, yet wishes to use the variable as one of the predictors for the synthesising model, then an empty (" ") method should be used while keeping the variable in `visit.sequence`. Note that variable/s to be synthesised later in `visit.sequence` can not be used as predictor/s for variable/s which appears before it. Though, variable/s can explicitly be removed as a predictor/s for any specific variable/s by updating the `predictor.matrix`. The `predictor.matrix` is a matrix with ones and zeros; Ones indicates that the variables should be used in the prediction model for generating synthetic values for a particular variable and zeros for otherwise.

3.1.4. Handling Data with Restricted or Missing Values

Relationship between variables can diversify significantly within a data set. Some variable can have a dependency on each other or could be tightly linked. As the goal of the synthetic data is to mimic all characteristics of the original data, these restrictions should be preserved during the data synthesis process. For example, in a medical data set, the variable containing information about the patient's sibling's medical history is restricted to the variable containing information whether the patient has siblings; This restriction needs to be addressed in order to get the best results out of the synthesis process. Simply when other variables determine the value for some case, the rule and corresponding values should be specified using `rule` and `rvalues` parameters. Furthermore, if the data set has missing values and the values are defined with something distinct than the `R` missing data code `NA`, it should be specified in `cont.na` parameter of the `syn()` function. Missing values in categorical variables are handled as additional categories. However, missing values in continuous variables are modelled in two steps. First, an auxiliary binary variable is synthesised to model whether a value is missing or not, and if there are multiple types of missing values, an auxiliary categorical variable is created to record this. Second, a synthetic model is fitted to non-missing values, and synthetic values are generated for non-missing categories in the auxiliary variable. Finally, the auxiliary variable, variable with non-missing values, and zeros for remaining records are used for prediction of other variables.

3.2. Utility Measures of Data

The purpose of a synthetic data set is to resemble all the properties of the original data set. Thus, analyses made on synthetic data set should lead to the same conclusions to the analyses made on the original data set. In theory, to achieve the formally mention purpose, the model used for the synthesis process should resemble the process of the original data generation. The methods to assess the utility of the synthetic data set can be broadly divided into two approaches: general utility and specific utility [40]. General utility assesses whether synthetic data have overall similarities in the statistical properties and multivariate relationships with the original data set. Whereas, specific utility assesses the similarity of performance of a fitted model on the synthetic data to the original data. The **Synthpop** package provides two types of analyses for the synthetic data set based on the general and specific utility of the data set utilising the `compare()` function in the package. First is the relative frequency distribution, and second is the linear machine learning model's confidence interval overlap. However, in this study, besides relative frequency distribution from the package, more rigorous analyses will be performed.

The overall utility of the synthetic data will be assessed on how adequately synthetic data succeed at all conducted utility tests. In order to succeed at a utility test, synthetic data need to resemble all the properties of the original data with at most statistically non-significant difference. For formal assessments, hypothesises will be as follows: Let D denote an original data set, and S_i denote a synthetic data set where i indicates the index for synthetic data produced with the different synthesising method. Let t

denote a vector of tests which returns a statistic, and C^* be a comparison function which returns a $p - value$. Finally, comparing the output of C^* with α, a threshold value for the level of significance.

$$H_o : C^*\{t(D), t(S_i)\} \geqslant \alpha, \qquad \text{for all } t \in [0, \tau]$$

$$H_a : C^*\{t(D), t(S_i)\} < \alpha, \qquad \text{for any } t \in [0, \tau]$$

The quality of the synthetic data will be estimated based on whether utility tests lead to failing to reject the null hypothesis. In order to fail to reject the null hypothesis, synthetic data must have $p - value$ larger or equal to α for all utility tests. The null hypothesis will be rejected if synthetic data possess $p - value$ smaller than α for any utility test leading to accept the alternate hypothesis. Note that the α is set to $0{,}05$ for all tests.

3.2.1. General Utility Measures

Data visualisation is the presentation of data in a visual or graphical format. A visual representation of data helps data analysts to process information from data faster than from written information. Visualisation of frequency distribution can reveal a lot about the data and its properties. Four principal characteristics of the frequency distribution are [41]:

1. The measure of central tendency and location (mean, median, mode)
2. The measure of dispersion (range, variance, standard deviation)
3. The extent of symmetry/asymmetry (skewness)
4. The flatness or peakedness (kurtosis)

On the other hand, relative frequency distribution provides the fraction or proportion of times a value occurs in data sets. A side-by-side univariate distribution of each variable in the synthetic and original data set will be plotted to compare the changes in the probability distribution, which can be used to determine the likelihood of specific results to occur within a given population [37]. Furthermore, the two-sample Kolmogorov–Smirnov test will be used to evaluate whether two underlying one-dimensional probability distribution differs in two different data sets (original and synthetic data set) for each variable. Since two data sets can possess nearly identical statistical properties, yet have very different distributions. In this case, the Kolmogorov–Smirnov statistic is:

$$D^* = \sup_X(|\hat{F}_1(X) - \hat{F}_2(X)|) \tag{1}$$

where $\hat{F}_1$ and $\hat{F}_2$ are the empirical distribution functions of the first and the second sample, respectively [42]. And sup is the supremum function. Moreover, the Cucconi test will be performed to evaluate whether the scale and location of the two data

set have a statistically significant difference by comparing the central tendency and variability.

Apart from visualising frequency distributions, visualisation of data points itself can help data analyst have a look at data from a different perspective. Visualisation of data directly, which has more than three dimensions is currently out of scope, but dimension reduction techniques which preserve the relationship between variables can be used as a pre-step. Uniform Manifold Approximation and Projection (UMAP) is a dimension reduction technique that can be used for visualisation similarly to T-distributed Stochastic Neighbor Embedding (t-SNE) [43], but also for general non-linear dimension reduction [44]. UMAP is constructed from Riemannian geometry and algebraic topology based theoretical framework. The result is a scalable algorithm that applies to real-world data. Despite being similar to t-SNE, it is competitive for visualisation quality and arguably preserves more of the global structure. Following the dimension reduction of the data, while preserving both global and local structures, data can be visualised in either two or three dimensions.

The bivariate Pearson product-moment correlation coefficient (PPMCC) is a parametric measure of the linear correlation between pairs of continuous variables. PPMCC produces a sample correlation coefficient, R, which measures the strength and direction of the linear relationship. The PPMCC also evaluates whether there is significant statistical evidence for a linear relationship among the same pairs of variables, represented by a population correlation coefficient, ρ (rho). The Pearson correlation between variables X and Y is calculated as:

$$\rho(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y}, \tag{2}$$

where cov is the covariance of variables X and Y, σ_x is the standard deviation of variable X, and σ_y is the standard deviation of variable Y. It is beneficial to note that a data set which does or does not have linear correlations, nevertheless, can have non-linear or complex correlations.

3.2.2. Specific Utility Measures

The specific utility of the data can be assessed by comparing the performance of the fitted synthetic and original models. In this thesis, multiple machine learning models were used as classifiers, such as Gradient Boosting Machine, Pattern Recognition Network, *k*-Nearest Neighbours, and Linear Discriminant Analysis. Needless to say, most of these models can also be used for regression. Different types of machine learning models were used to evaluate the generality of the primary method of synthesis **"Synthpop"**. Moreover, the performance of the fitted model will be examined on multiple parameters for overall performance estimation.

Gradient Boosting Machine

Boosting algorithms were initially introduced by the machine learning community [45, 46, 47] for classification problems. The principle approach of the boosting algorithms is to combine several simple models iteratively, termed weak learners to

obtain a strong learner with improved predictive accuracy. A new statistical point of view for boosting was introduced to connect the boosting algorithm to the concept of loss functions [48]. Later, an extended boosting algorithm for regression termed Gradient Boosting Machine (GBM) was introduced [49]. The GBM is similar to a numerical optimisation algorithm that aims to find an additive model that minimise the loss function. Thus, GBM is a classification and regression forward learning ensemble technique, which generates a prediction model in the form of an ensemble of weak prediction models, typically decision trees that best reduces the loss function. This study follows the GBM algorithm implemented in H2O package in R [50], which follows the algorithm specified by Hastie et al. [51 p. 359-360]. The function $f_0(x)$ is estimated by minimising loss function L over the training data set.

Algorithm 1. Gradient Boosting Machine

1 Initialize $f_0(x) = argmin_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$

2 For $m = 1$ to M:

3 (a) For i = 1,2,. . . ,N compute

4

$$r_{im} = - \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f=f_m-1}$$

5 (b) Fit a regression tree to the target r_{im} giving terminal regions R_{jm}, j = 1,2,. . . , j_m.

6 (c) For j = 1,2,. . . , J_m compute

7

$$\gamma_{im} = argmin_\gamma \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma)$$

8 (d) Update $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{j_m} \gamma_{jm} I(x \in R_{jm})$

9 **Output**: $\hat{f}(x) = f_m(x)$;

The index for the weak learner added to the ensemble is denoted by m and M is the maximum number of iterations. For each weak learner j, the residuals γ_{im} are computed and a regression tree is fitted. Finally, the current model is added to the fitted regression tree to improve the overall accuracy of the final model.

Pattern Recognition Network

Artificial Neural Networks (ANN) are computing systems inspired by biological neural networks. The primary goal of the neural network method was to solve problems similarly that a human brain would. Neural Networks (NN) for pattern recognition is an advancing field. The NN determines the appropriate mathematical use to turn the input into output, whether they have a linear or non-linear relationship. For each input, the network moves through the layers calculating the probability of each output. Pattern recognition networks are feedforward networks that can be trained to classify inputs according to target classes. The target data for pattern recognition networks should consist of vectors of all zero values except for a "1" in element i, where i is the class they are to represent. Pattern recognition network used in this study is a two-layer feedforward network with tan-sigmoid transfer function in

the hidden layer and a softmax transfer function in the output layer [52]. The function for the pattern recognition network (Figure 5) in matrix notation is:

$$y = f(x) = G(b^{(2)} + W^{(2)}(s(b^{(1)} + W^{(1)}x))), \tag{2}$$

where $b^{(1)}$ and $b^{(2)}$ are bias vectors, $W^{(1)}$ and $W^{(2)}$ are weight matrices, and finally, s and G are transfer functions for tan-sigmoid and softmax respectively.

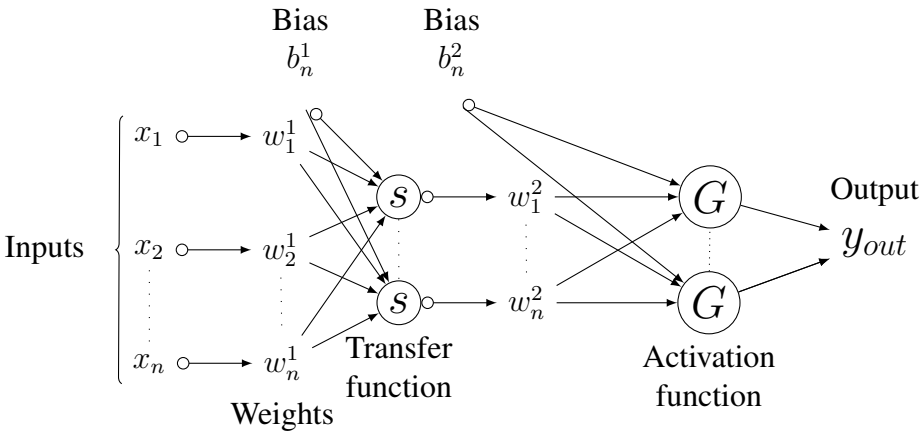
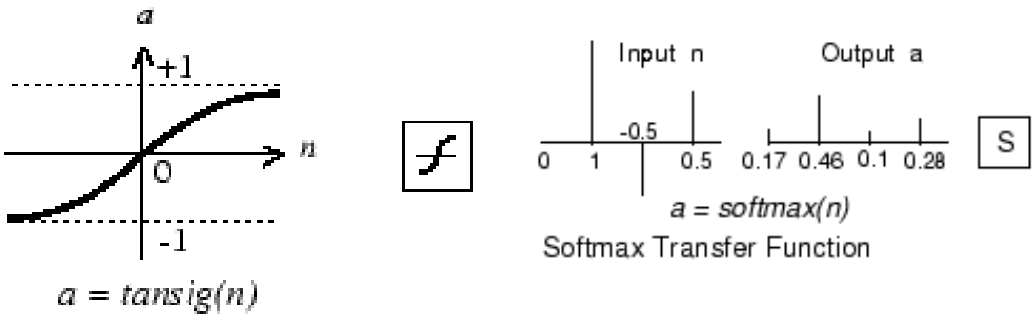


Figure 5. Pattern recognition network



(a). Hyperbolic tangent sigmoid transfer function

(b). Soft max transfer function

Figure 6. Transfer functions [52]

The tan-sigmoid transfer function (Figure 6(a)) is calculated as:

$$s = tansig(n) = 2/(1 + exp(-2 * n)) - 1, \tag{3}$$

And the softmax transfer function (Figure 6(b)) is calculated as:

$$G = softmax(n) = exp(n)/sum(exp(n)), \tag{4}$$

Where n is S-by-Q matrix of net input (column) vectors.

***k*-Nearest Neighbours**

The k-Nearest Neighbours (k-NN) is a non-parametric method used for regression and

classification. For both classification and regression, the input consists of *k* closest training examples in the feature space. The output in classification problem using *k*-NN is a class association. Each sample is classified based on a plurality vote of its neighbours and is assigned to the class most common among its *k* nearest neighbours. The *k* is a positive integer, usually small. The *k*-NN algorithm stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). In this study, *k*-nearest neighbour approach for classification with, for simplicity, Euclidean distance in feature space is implemented [53]. Let X and Y represent the feature vectors $X = (x_1, x_2, \ldots, x_m)$ and $Y = (y_1, y_2, \ldots, y_m)$, where m is the dimensionality of the feature space. To calculate normalised Euclidean distance between A and B, the metric is:

$$dist(X, Y) = \sqrt{\frac{\sum_{i=1}^{m}(x_i - y_i)^2}{m}} \tag{5}$$

Linear Discriminant Analysis

In pattern recognition, Linear discriminant analysis (LDA) is a method used to find a linear combination of features that characterise or discriminate two or more classes of events [54]. LDA is a classification method in which it assumes that distinct classes generate data based on different Gaussian distribution. In order to generate or train a classifier, the fitting function estimates the parameters of Gaussian distribution for each class; the model has the same covariance matrix for each class, only the means vary. Whereas, to predict classes of new data using a trained discriminant classifier, it finds the class with the smallest misclassification cost.

$$\hat{y} = \underset{y=1,\ldots,K}{argmin} \sum_{k=1}^{K} \hat{P}(k|x)C(y|k), \tag{6}$$

where $\hat{y}$ is the predicted classification while minimising the expected classification cost, the number of classes is denoted by K, the posterior probability of class k for observation x is denoted by $\hat{P}(k|x)$, and the cost of classifying an observation as y when its actual class is k is $C(y|k)$ [53].

The LDA model used in this study uses class empirical probabilities in Y as the class prior probabilities. Where Y is the class labels and Each row of Y represents the classification of the corresponding row of X. The matrix X contains predictor values in the form of a numeric matrix. With its column representing one variable, and row is representing one observation. If the prior probability of class k is represented by $P(k)$. Then the posterior probability of class k for observation x is:

$$\hat{P}(k|x) = \frac{P(x|k)P(k)}{P(x)} \tag{7}$$

Performance evaluation parameters

In order to evaluate the performance of machine learning models, precisely the

problem of statistical classification, the parameters applicable are based on the elements from a special kind of contingency table termed confusion matrix or error matrix [55]. The confusion matrix consists of terms such as True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN), these terms are used to compare the label of classes as shown in Table 2. TP is a Positive review classified as positive, and TN is a negative review classified as negative. Whereas FP is a negative review classified as positive, and FN is a positive review classified as negative.

Table 2. Confusion Matrix

		Actual Values	
		Positive	Negative
Predicted values	Positive	TP	FP
	Negative	FN	TN

Based on the values obtained from the confusion matrix, parameters such as precision, recall, F-measure and accuracy can be calculated to evaluate classifier's performance.

Precision: It is the number of correct positive values divided by the number of all positive predicted values.

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

Recall: It is the number of correct positive values divided by the number of all positive actual values.

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

Accuracy: It is the number of correct values divided by the total number of the returned values.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

F_1 score: It is the harmonic mean of precision and recall, and it is most valuable when there is an uneven class distribution (a large number of true negative) [56 p. 27]. Another essential information to remark about the F1 score is the process of handling the cost of the false positive and false negative predictions—the F1 score emphasis more on lowering these costs. In a classification problem, when one class is more critical than the other, the weight of the cost of false-positive and false-negative can vary. For example, in cancer detection, the cost of having higher false-positive predictions is lower than the cost of having higher false-negative predictions.

$$F_1 score = 2 * \left[\frac{Precision * Recall}{Precision + Recall} \right] \tag{11}$$

Feature Importance and Accumulated Local Effects
In order to estimate the importance of a feature for predictions and order of importance, Feature Importance (FI) plots can be used. FI measure is calculated by the increase of the model's prediction error after permuting the features. A feature is said to be "important" if permuting its value increases the model error, as the model relied on the feature for predictions. On the other hand, a feature is said to be "unimportant" if permuting its value does not change the model error, as the model did not rely on the feature for predictions [39, 57]. It is necessary to note that the feature importance is highly dependent on the data set. FI can differ every time it is calculated as the order of selection of the features is random. Furthermore, Accumulated local effects (ALE) plots show the average model prediction over the feature. In other words, the ALE plot shows the effect of a feature for prediction. ALE plots are computed as accumulated differences over the conditional distribution and partial dependence over the marginal distribution. Similarly to the FI, ALE plots are also highly dependent on the data set; it can differ depending upon whether it is calculated over a testing set or a subset of a particular group.

3.3. Quality of Information Content

The goal of data anonymisation procedure is to reduce semantics, meaning minimising or removing personal information in a data set [58, 59]. Data anonymisation can cause distortion and information loss in the data set. In this section, the concepts of information theory will be used, to quantify the level of distortion and information loss. Concepts such as evaluating change in entropy and estimating the mutual information (MI) between variables will be used [59, 60].

3.3.1. Entropy

Entropy is a fundamental quantity in information theory associated with any random variable. Entropy can be interpreted as the level of information, surprise, or uncertainty associated with the value of a random variable or the result of a random process [61]. The bit, which is the unit of entropy, is adopted as a quantitative measure of information, or measure of surprise. The entropy of a random variable X, with possible outcomes x_i, each with a probability of occurrence $P_X(x_i)$ is calculated as:

$$H(X) = -\sum_i P_X(x_i) log_b P_X(x_i) \tag{12}$$

The entropy is maximum when all outcomes are equally likely in a system. If the system moves away from equally likely outcomes or introduces some predictability, the entropy goes down. The fundamental idea of the information theory is that, if the entropy of an information source or system or data set drops, that means fewer questions are needed to ask to guess the outcome. Entropy is directly proportional to uncertainty, i.e., as the value of entropy increases due to unpredictability, uncertainty in the system's outcome increases and the ability to compress decreases, similarly, if

the value for entropy decreases due to known structure, then the ability to compress increases, which lead to entropy being indirectly proportional to the ability to compress.

3.3.2. Mutual Information

The MI is a measure of mutual dependence between two random variables. MI measures the information gain for a random variable X when information about another random variable Y is given. MI between two random variables X and Y can be calculated as:

$$I(X,Y) = \sum_{x_i \in X, y_i \in Y} p(x_i, y_i) log(\frac{p(x_i, y_i)}{p(x_i)p(y_i)}) \tag{13}$$

or

$$I(X;Y) = H(Y) - H(Y|X) \tag{14}$$

If entropy $H(Y)$ is a measure of uncertainty about a random variable Y, then $H(Y|X)$ is a measure of what X does not say about Y. In other words, $H(Y|X)$ is the amount of uncertainty remaining about Y after X is known. Therefore, the equation can be interpreted as the amount of uncertainty in Y, minus the amount of uncertainty in Y after X is known. Furthermore, this provides the inherent meaning of MI as the amount of information or reduction in uncertainty that one random variable provides about the other.

The Kraskov's estimator [60] of mutual information is closely related to Shannon's entropy, but Kraskov's estimator relies on the count of nearest neighbours. Kraskov's estimator, along with many others [62], uses canonical distance defined in metric space for computability over Euclidean space and uses Euclidian distance as the distance function. The mutual information estimator $I^{(2)}$ between two random variables $\mathbf{x}_i$ and $\mathbf{y}_i$ is defined as:

$$I^{(2)}(X,Y) = \Psi(k) - 1/k - < \Psi(\mathbf{n}_x) + \Psi(\mathbf{n}_y) > + \Psi(N), \tag{15}$$

with Ψ the digamma function and k denoted the number of neighbours. Where $< \cdots >$ denotes the averages of both vectors $\mathbf{n}_x(i)$ and $\mathbf{n}_y(i)$ holding counts of neighbours over all $i \in [1,\ldots,N]$ and over all realizations of the random samples.

In this study, a variation of the second algorithm from Kraskov's estimator proposed by Oliver et al. [59] to use the method over non-Euclidean spaces using non-Euclidean distances will be used. Where the calculation requires the nearest neighbours of points in joint space and counting how many lie in an absolute ball.

3.4. Data Sets

This section outlines the structure and objective of each data set used in this study. The purpose of selecting two different types of data sets is to evaluate the comprehensiveness and implementation of the primary tool of synthesis **"Synthpop"**.

3.4.1. DIPP Data Set

Finland has the highest incidence of Type 1 Diabetes (T1D) in the world amongst young children, currently standing at approximately 72 in every 100,000 children under the age of 15 years [63]. The Type 1 Diabetes Prediction and Prevention (DIPP) Study was established in 1994 in three university hospitals in Finland to understand/learn the pathogenesis of T1D [15]. The goal of this ongoing study is to find new treatments and preventative methods by assessing risk factors in the development of T1D. The DIPP study is a population-based long-term clinical follow-up study that consists of screening newborns for increased genetic risk for diabetes.

The DIPP database used in this study has been collected since 1994 only at the Oulu University Hospital and contains information from over 6500 subjects in the form of longitudinal data; recorded since the birth of the subject. The database includes information about the subject along with the monitoring information of siblings and parents. The database also suffers from missing values due to non-standardised input methods such as information entered by hands during collection. The database comprises variables such as blood samples, infections, medications, vaccines, nutrition, and environmental factors. Blood sample data includes three autoantibody values of glutamic acid decarboxylase (GADA), protein tyrosine phosphate autoantibody (IA2A), and antibodies of insulin (IAA).

The data set used in this study was built and pre-processed from the original DIPP database and modelled by M.Sc. Oana Maria Stoicescu and most of the variables of the data set used in this is study can be referred to her thesis [64]. The data until the age of 12 months was aggregated to utilise information gain from that data to predict the positivity of the autoantibodies later in life. First, variables such as infections were aggregated to value *0* if the number of infections is zero or to value *1* if more than one or two infections in the first 12 months of age. Infections leading to hospital care and other similar variable were cumulated similarly. Furthermore, for variables such as autoantibodies, the maximum autoantibody value was taken into account before the first positive value before 12 months of age occurred. Later excluded the seven subjects whose autoantibodies values were in positive range before 12 months of age due to autoantibodies transmitted from mother. Finally, a response variable `"POS_antibodies"` was defined based on the positivity of autoantibodies. Class negative, if the subject never had an occurrence of positive value in any autoantibodies up until 170 months of age and class positive, if the subject had two or more consecutive positive value occurrences in any autoantibodies up until 170 months of age. The value of an autoantibody is positive if they are higher than a specific threshold for the respective autoantibodies. The threshold values for GADA, IA2A, and IAA are 5.34, 0.42, and 3.47, respectively. Overall, providing 30 attributes using a small subset of data of 1329 subjects. Out of which 839 subjects belong to the positive class and

490 to the negative class. Table 3 provides the list of all attributes in the data set and their description. The goal of the data set is to predict the probability of the positivity of autoantibodies before the age of 15 years by utilising information gain from the first 12 months of data.

3.4.2. HAR Using Smartphone Data Set

Recognising and understanding human behaviour using computational artefacts and efforts of applying that information to improve current Human-Centered Computing (HCC) is an emerging research field [65]. Human Activity Recognition (HAR) has also demonstrated to be a significant source of knowledge by accurately identifying human activities for better patient recovery training guidance and could send an early alarm of emergencies such as a stroke or a fall [66]. Combining information gained from a HAR system with other information such as heart rate from a biometric monitoring device can consolidate, for example, the clinical or laboratory investigation and diagnoses and its treatment. The HAR data set used in this study is an open data set, which implies that the data is free and available at the University of California Irvine machine learning repository [16]. The primary intention of the HAR data set collectors was to build models targeting the recognition of six different human activities using smartphones accelerometers and gyroscopes. Moreover, it was made publicly available by being inspired by many other researchers or data collectors of the similar research field.

The motivation of using HAR data set in this study is to utilise a different, big in size, and more complex data set consisting of a rather high correlation between variables. The data set features were derived from raw signals using similar variables which cause the high correlation within the data set. Furthermore, the data set is openly available, so the findings from this study can be replicated or further questioned, which supports the main objective of the study. Therefore, this study examines the performance of the data synthesis tool over the HAR data set towards the possibility of data sharing for similar data sets.

The data set was collected from 30 subjects performing Activities of Daily Living (ADL). The Age group of subjects was 19 to 48 years old, and each subject performed six different activities, including walking, walking upstairs, walking downstairs, sitting, standing, and lying while wearing the Samsung Galaxy S II smartphone on the waist. From the smartphone's accelerometer and gyroscope, 3-axial raw signals "tAcc-XYZ" and "tGyro-XYZ" at a constant rate of 50Hz were recorded (prefix 't' to denote time). Figure 7 represents the workflow of the generation of the signals for feature extraction.

The raw signals collected using the smartphone's accelerometer and gyroscope sensors were pre-processed for noise reduction using a median filter and a 3rd order low-pass Butterworth filter with a corner frequency of 20 Hz to sample in a fixed-width sliding window of 2.56 seconds with a 50% overlap reaching 128 samples per window. Furthermore, the body acceleration signal "tBodyAcc-XYZ" and gravity acceleration signals "tGravityAcc-XYZ" were acquired using another low-pass Butterworth filter with a cutoff frequency of 0.3 Hz for separating gravitational and body motion component from the acceleration signal. The body linear acceleration (tBodyAcc-XYZ) and angular velocity (tBodyGyro-XYZ)

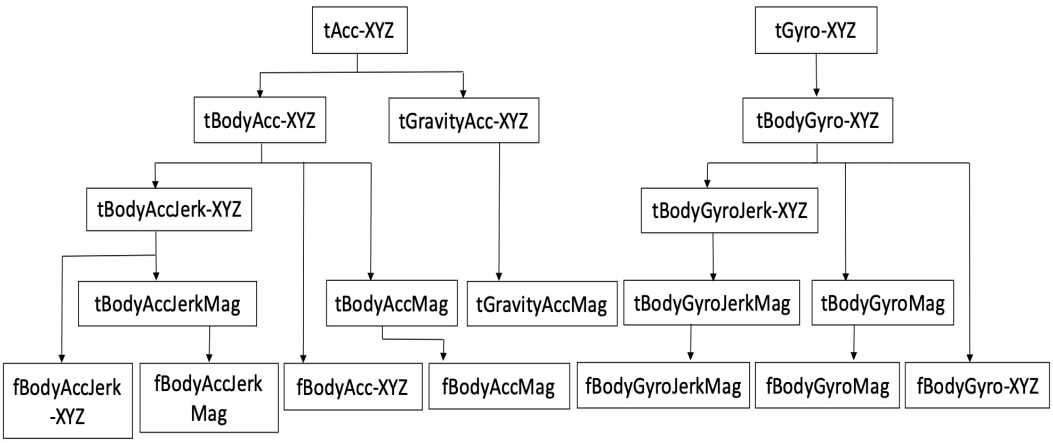


Figure 7. Signal processing for feature extraction

were derived in the time domain to obtain Jerk signals (tBodyAccJerk-XYZ and tBodyGyroJerk-XYZ). Additionally, the magnitude of these three-dimensional signals was calculated using the Euclidean norm (tBodyAccMag, tGravityAccMag, tBodyAccJerkMag, tBodyGyroMag, tBodyGyroJerkMag). Finally, a Fast Fourier Transform was implemented to some of these signals to generate "fBodyAcc-XYZ", "fBodyAccJerk-XYZ", "fBodyGyro-XYZ", "fBodyAccMag", "fBodyAccJerkMag", "fBodyGyroMag", "fBodyGyroJerkMag". (the prefix 'f' to indicate frequency domain signals).

Table 4 list all the variables used for calculating the features from the signals. Table 5 lists all the signals used for feature extraction, including the list of variables used for extraction, ordered according to their occurrence in the data set. To denote the 3-axial signals in the X, Y and Z directions, "-XYZ" is used in the signal naming. From few selected signals, supplementary vectors were obtained by averaging the signals (used later for angle variable from Table 4): gravitymean, tBodyAccMean, tBodyAccJerkMean, tBodyGyroMean, tBodyGyroJerkMean.

Table 6 list last seven features of the data set calculated using the angle between two vectors (Variable 17 in Table 4). Finally, 561 feature vectors were constructed for all six ADL samples for each subject. The complete list of variables of each feature vector is available in *'features.txt'* file at the University of California Irvine machine learning repository [16].

Table 3. Names and description of attributes

Order	Attributes	Description
1	POS_antibodies	Response variable - 1 the child had 2 or more consecutive positive samples in any of the autoantibodies, otherwise 0.
2	length	Length at birth (cm)
3	weight	Weight at birth(g)
4	circle_of_head	Head circumference measured at birth (cm)
5	ratio_head_length	Ratio between head circumference and length measured at birth (cm)
6	Mom_birth_age	Age of mother at the time of birth (years)
7	height_growth	Growth rate calculated by : (height measured in the last visit - length at birth)/Age in months
8	weight_growth	Growth rate calculated by : (weight measured in the last visit - birth weight/1000)/Age in months
9	GADA.UUSI	Maximum value of GADA antibody that occurred before 12 months old (negative value)
10	mIAA.3.470	Maximum value of IAA antibody that occurred before before 12 months old (negative value)
11	IAA.0.42	Maximum value of IA2A antibody that occurred before before 12 months old (negative value)
12	s.gender	Gender 1 - male, 2 - female
13	duration	Pregnancy duration: 0 - pre term 0 to 37 weeks, 1 - normal 37 to 42 weeks, 2 - post term > 42 weeks
14	month	Month of birth - from 1 to 12
15	mother_antib	1 - if the child's mother had positive autoantibodies, 0 otherwise
16	sibling_antib	1 - if the child's sibling had positive autoantibodies, 0 otherwise
17	has_sibling	1 - if the child has siblings, 0 otherwise
18	is_mom_t1d	Does mom have t1d 1 - yes, 0 - no
19	is_dad_t1d	Does dad have t1d 1 - yes, 0 - no
20	v.breastfeeding_only	Age when exclusive breastfeeding has ended (months)
21	v.breastfeeding_ended	Age when any breastfeeding has ended (months) - maximum is 12 which means currently still breastfeeding.
22	i.infections_ear	0 - no ear infections, 1 - 1 infection, 2 - more than 2 infections
23	i.infections_eye	0 - no eye infections, 1 - more than 1 infections
24	i.infections_hospital_care	0 - no infections requiring hospital stay, 1 - more than 1 infections
25	i.infections_airway	0 - no airway infections, 1 - 1 infection, 2 - more than 2 infections
26	i.infections_gastric	0 - no infections, 1 - more than 1 infections
27	i.infections_other	
28	i.infections_fever	
29	i.infections_roseola	
30	i.infections_chickenpox	

Table 4. List of variable used for feature extraction with description

Order	**Variable**	**Description**
1	mean()	Mean value
2	std()	Standard deviation
3	mad()	Median absolute deviation
4	max()	Largest value in array
5	min()	Smallest value in array
6	sma()	Signal magnitude area
7	energy()	Energy measure. Sum of the squares divided by the number of values.
8	iqr()	Interquartile range
9	entropy()	Signal entropy
10	arCoeff()	Autorregresion coefficients with Burg order equal to 4
11	correlation()	correlation coefficient between two signals
12	maxInds()	index of the frequency component with largest magnitude
13	meanFreq()	Weighted average of the frequency components to obtain a mean frequency
14	skewness()	skewness of the frequency domain signal
15	kurtosis()	kurtosis of the frequency domain signal
16	bandsEnergy()	Energy of a frequency interval within the 64 bins of the FFT of each window.
17	angle()	Angle between to vectors.

Table 5. Summarised list of all the signals used for feature extraction with the variables used. In order corresponding to their occurrence in the data set. *variable represents the order number in Table 4. **detailed list of features using angle in Table 6

Order	**Signal**	**Variables***	**Number of features**
1	tBodyAcc-XYZ	1 to 11	40
2	tGravityAcc-XYZ	1 to 11	40
3	tBodyAccJerk-XYZ	1 to 11	40
4	tBodyGyro-XYZ	1 to 11	40
5	tBodyGyroJerk-XYZ	1 to 11	40
6	tBodyAccMag	1 to 10	13
7	tGravityAccMag	1 to 10	13
8	tBodyAccJerkMag	1 to 10	13
9	tBodyGyroMag	1 to 10	13
10	tBodyGyroJerkMag	1 to 10	13
11	fBodyAcc-XYZ	1 to 9 and 12 to 16	79
12	fBodyAccJerk-XYZ	1 to 9 and 12 to 16	79
13	fBodyGyro-XYZ	1 to 9 and 12 to 16	79
14	fBodyAccMag	1 to 9 and 12 to 15	13
15	fBodyAccJerkMag	1 to 9 and 12 to 15	13
16	fBodyGyroMag	1 to 9 and 12 to 15	13
17	fBodyGyroJerkMag	1 to 9 and 12 to 15	13
18	Angle**	17	7
		Total	**561**

Table 6. List of last seven features in the data set obtained using angle variable from Table 4 with corresponding vectors. *Index represents the actual index of feature in the data set.

Index*	**Feature**
555	angle(tBodyAccMean,gravity)
556	angle(tBodyAccJerkMean),gravityMean)
557	angle(tBodyGyroMean,gravityMean)
558	angle(tBodyGyroJerkMean,gravityMean)
559	angle(X,gravityMean)
560	angle(Y,gravityMean)
561	angle(Z,gravityMean)

4. EXPERIMENTS AND RESULTS

This chapter reports a comprehensive analysis of the primary tool of data synthesis. The tool has been applied to two different data sets: DIPP and HAR data set. The chapter aims to evaluate whether and to what extent the data synthesis process preserves the general and specific utility along with the quality of the information content of the original data set. First, the performance of the different methods of synthesis is evaluated based on the specific utility of the synthetic data sets to select the fittest method of synthesis. Specific utility compares the performance of the synthetic and original data sets over corresponding data-fitted models and illustrates a visual model comparison based on the FI and their ALE for model fitting. Following the method selection, general utility and the quality of information content is assessed for the selected synthetic data set. The general utility examines the statistical properties of the synthetic data set to the original data set based on the correlation between data variables, data visualisation, data distributions, and data similarity. Whereas, the quality of the information content is measured from an information-theoretic point of view covering entropy and MI within the data sets. Similarly, all three primary analyses are repeated for the HAR data set with the same motivation for general utility and quality of information contained in the data set analyses; however, with additional motivations for specific utility experiments. For HAR data set, the specific utility also measures whether the size of data during data synthesis process affects the performance along with the relevancy of the data synthesis tool for secondary data analysis.

4.1. DIPP Data Set

The pre-processed version of the DIPP data set is a data frame with 30 attributes, including the response variable for 1329 subjects in total. Multiple variables in the data set were first turned into factors using `as.factor()` command in R. Later, the data set was synthesised numerous times via `syn()` command from **Synthpop** package using several methods. As mentioned earlier in Subsection 3.1.2, the first variable to be synthesised in the data is by default generated using `"sample"` method. In our case, the response variable `"POS_antibodies"` is the first variable to be synthesised, and then the rest of the attributes. Table 3 provides the list of all attributes in the data set, and their description in the order of synthesis, i.e., `"visit.sequence"`.

Methods of synthesis

Both non-parametric and parametric methods of synthesis were used in the engendering of the synthetic data sets. Table 7 lists the methods used for generating the corresponding synthetic data with denoting names and method description. Table 8 lists all attributes and corresponding `"parametric"` method applied. For all non-parametric method, every attribute was synthesised using same method. Each synthetic data set was generated using `seed` value for result replication. A total of 5 synthetic data sets were generated for initial experimentation using 5 different methods (*SynD1 to SynD5*). One method of synthesis which performs the best out of

those 5 methods was selected for generating another synthetic data set by setting the argument `proper` to `TRUE` for proper synthesis for further analysis (*SynD6*).

Table 7. Denoted names for synthetic data sets and methods used for creation, *List of parametric method for each variable is listed in Table 8.

Synthetic data	Method	Description
SynD1	`"cart"`	classification and regression tree
SynD2	`"ctree"`	classification tree
SynD3	`"rf"`	random forest
SynD4	`"bag"`	bagging
SynD5	`"parametric"`	parametric* method to each variable based on their data type
SynD6	`"cart"`	classification and regression tree with `proper` set to `TRUE`

Table 8. Attributes and parametric methods applied for generating *SynD5* data set.

Attributes	Method	Attributes	Method
POS_antibodies	`"sample"`	sibling_antib	`"logreg"`
length	`"normrank"`	has_sibling	`"logreg"`
weight	`"normrank"`	is_mom_t1d	`"logreg"`
circle_of_head	`"normrank"`	is_dad_t1d	`"logreg"`
ratio_head_length	`"normrank"`	v.breastfeeding_only	`"normrank"`
Mom_birth_age	`"normrank"`	v.breastfeeding_ended	`"normrank"`
height_growth	`"normrank"`	i.infections_ear	`"polyreg"`
weight_growth	`"normrank"`	i.infections_eye	`"logreg"`
GADA.UUSI	`"normrank"`	i.infections_hospital_care	`"logreg"`
mIAA.3.470	`"normrank"`	i.infections_airway	`"polyreg"`
IAA.0.42	`"normrank"`	i.infections_gastric	`"logreg"`
s.gender	`"logreg"`	i.infections_other	`"logreg"`
duration	`"polyreg"`	i.infections_fever	`"logreg"`
month	`"normrank"`	i.infections_roseola	`"logreg"`
mother_antib	`"logreg"`	i.infections_chickenpox	`"logreg"`

4.1.1. Specific Utility

In this section, we evaluated whether different methods of synthesis preserve the specific utility of the original data set differently, after which we selected one method of synthesis that performs best out of all methods used. The goal is twofold, first to investigate if synthetic data sets can be used for machine learning problems when the original data can not be acquired and second to assess how well synthetic data sets perform on the machine learning classifier as compared to the original data set.

The machine learning classifier used is the GBM model, which was fitted, validated, and tested 10 times (for more stable performance of the model) with all data sets from Table 7 along with the original data set, each time with different `seed` value. Additionally, each data set was divided into three splits with different `seed` value, before model fitting, 75.0% of data for training, 12.5% for validation, and 12.5% for testing.

Comparing different methods

We compared the results obtained from synthetic data test sets to the results of the original data test set; to evaluate which synthesising method produces the synthetic data set principally resembling the performance of the original data set. The performance measure used is confusion matrix and parameters derived from it. The motivation behind using multiple performances evaluate parameters is to provide a more robust interpretation [67], as a model can have very high accuracy, yet suffer from low precision [68 p. 128-129].

Table 9. Data sets and their performance over GBM model

Data set	**Confusion Matrix**			**Evaluation Parameter**		**Accuracy**
Original Data		Predicted labels		F1 score	Area Under ROC	0.87
		Negative	Positive			
	Negative	89	16	0.85	0.95	
	Positive	5	56	0.82		
SynD1		Predicted labels		F1 score	Area Under ROC	0.88
		Negative	Positive			
	Negative	83	19	0.88	0.93	
	Positive	1	63	0.85		
SynD2		Predicted labels		F1 score	Area Under ROC	0.86
		Negative	Positive			
	Negative	82	20	0.87	0.93	
	Positive	3	61	0.82		
SynD3		Predicted labels		F1 score	Area Under ROC	0.90
		Negative	Positive			
	Negative	89	13	0.91	0.95	
	Positive	4	60	0.87		
SynD4		Predicted labels		F1 score	Area Under ROC	0.93
		Negative	Positive			
	Negative	90	12	0.92	0.97	
	Positive	0	64	0.89		
SynD5		Predicted labels		F1 score	Area Under ROC	0.88
		Negative	Positive			
	Negative	98	4	0.85	0.92	
	Positive	15	49	0.78		

One sample out of ten for the comparative performance of synthetic data sets for all selected synthesis methods with the original data set can be seen in Table 9. Note

that the process was repeated 10 times for all data sets to perform a significance test over testing accuracies. The accuracies of each synthetic data set fitted model and the accuracies of the original data set fitted model were compared using C^* comparison function. The C^* function returns a $p-value$, table 10 provides the $p-value$ for each data set. Every single $p-value$ was calculated using t-test, comparing accuracies of every synthetic data set to the original data set over the GBM model, 10 iterations.

Table 10. Data set and p-values for their comparative accuracy with original data over GBM model *(fitted 10 times)*

Data set	P-value
SynD1	0.0965496
SynD2	0.0485093
SynD3	0.0026730
SynD4	0.0288157
SynD5	0.1755973

The objective is to fail to reject the null hypothesis, i.e., the difference in the performance of the synthetic data should differ with the performance of the original data with at most non-significant difference. In other words, aiming that the synthetic data set produced using any method does not need to perform better or should not perform worst than the original data, but it needs to perform as close as possible to the original data set. From Table 10, the data sets produced using method `"cart"` (*SynD1*) and `"parametric"` (*SynD5*) are only two data sets with $p-value$ greater than α, whereas rest have $p-value$ smaller than α. If the $p-value$ is greater than α, it states that according to statistics, these two data sets fail to reject the null hypothesis meaning that difference is statistically non-significant. Moreover, from Table 9, we can say that *SynD1* performs better than *SynD5* when other evaluation parameters are considered. As the overall performance difference to original data is smaller for *SynD1* as compared to *SynD5*. Note that the $p-value$ for each data set is calculated only using the accuracies of the model over the test set, which reflects the generalisability of the model.

Comparing simple and proper synthesis

Next, the original data set was synthesised again using `"cart"` method, but this time with setting the `proper` argument to `TRUE` (*SynD6*) for posterior predictive distribution of parameters (proper synthesis). Furthermore, a test set of original data sets is fed in the synthetic data fitted model to evaluate the level of local and global structure-preserving capacity of the synthesis method, and pertinence of one aspect of the secondary data analysis. The comparative performance can be seen in Table 11.

The $p-value$ from t-test for accuracies of *SynD6* data set in comparison to the accuracies of original data set is *0.0006553*. The $p-value$ is smaller than α, which suggests strong evidence to reject the null hypothesis. Furthermore, from the results shown in Table 11, we can use the outcomes to leverage the formal finding. The original data test set performs better for *SynD1* than for *SynD6* data fitted model. Since the difference in F1-score of original test data set is smaller for *SynD1* data fitted model than of *SynD6* data set fitted model.

Table 11. Data sets and their performance over GBM models

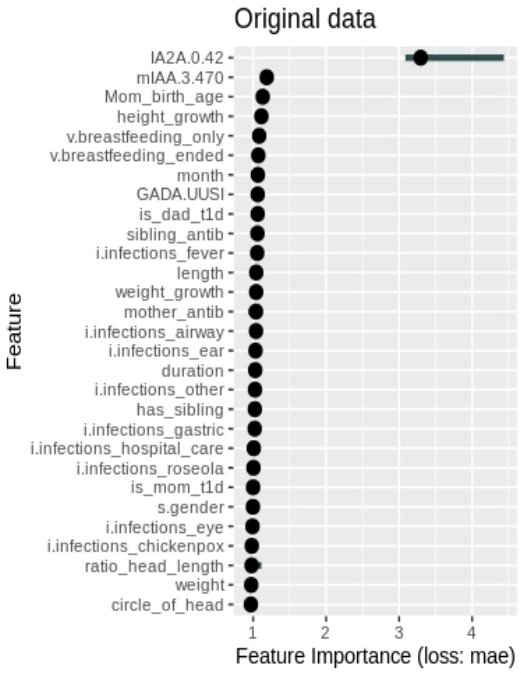
Test set	Model	Confusion Matrix			Evaluation Parameters		Accuracy
Original	**Original**		Predicted labels		F1 score	Area Under ROI	0.87
			Negative	Positive			
		Negative	89	16	0.82	0.95	
		Positive	5	56	0.85		
SynD1	**SynD1**		Predicted labels		F1 score	Area Under ROI	0.88
			Negative	Positive			
		Negative	83	19	0.85	0.93	
		Positive	1	63	0.88		
SynD6	**SynD6**		Predicted labels		F1 score	Area Under ROI	0.89
			Negative	Positive			
		Negative	83	18	0.86	0.95	
		Positive	0	65	0.89		
Original	**SynD1**		Predicted labels		F1 score	Area Under ROI	0.86
			Negative	Positive			
		Negative	92	13	0.83	0.96	
		Positive	6	55	0.87		
Original	**SynD6**		Predicted labels		F1 score	Area Under ROI	0.85
			Negative	Positive			
		Negative	80	25	0.79	0.93	
		Positive	0	61	0.87		

Visual model comparison

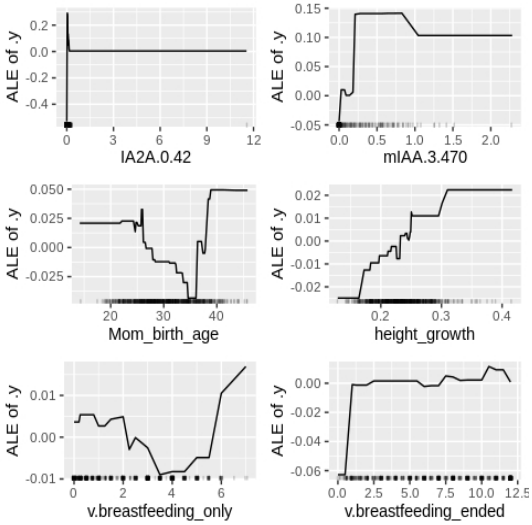
Next, we examined the FI plots for original, *SynD1*, and *SynD6* data sets and ALE plots of first 6 important features for each data set. The number of repetitions was set to 5, defining how often to shuffle features while calculating FI for more stable and accurate results. Figures 8, 9, and 10 show the FI and ALE plots for the original, *SynD1*, and *SynD6* data sets, respectively.

For original data (Figure 8) FI plot shows that the IA2 antibody values have the most substantial influence in the prediction of the positivity of the autoantibodies later in life, followed by the IAA antibody values, mother's age at the time of birth, height growth rate, age when exclusive breastfeeding ended, and age when any breastfeeding ended. From the ALE plots, we can interpret that after the IA2 value reaches a specific value, the probability of positivity reaches a constant, whereas it decreases with higher IAA value. If the mother's age at the time of birth is higher than approximately 37 years, the probability of positivity increases.

From the FI and ALE plots of original (Figure 8), *SynD1* (Figure 9), and *SynD6* (Figure 10) data set fitted models, we can interpret that *SynD1* data fitted model have higher number of same variables in the first 6 important features and their influence for the model prediction to the original data set fitted model as compared to the *SynD6* data fitted model. As the FI plots for original data set and *SynD1* have same first 3 important features and their ALE plots shows similar accumulated local effect for the matching features including feature `"v.breastfeeding_only"`.

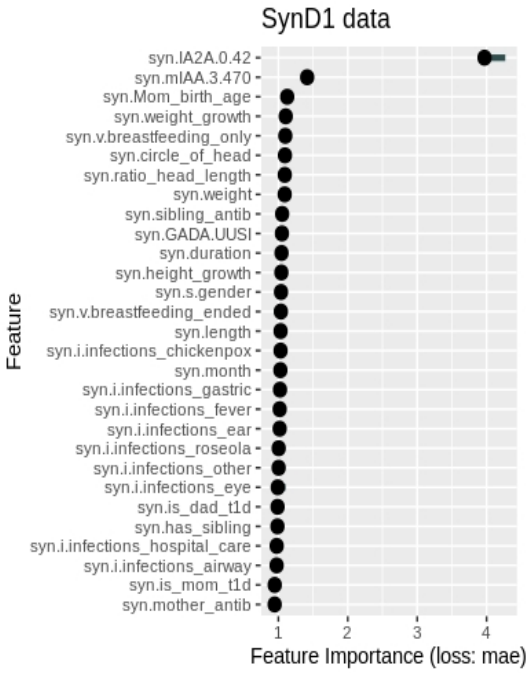


(a). Feature importance plot

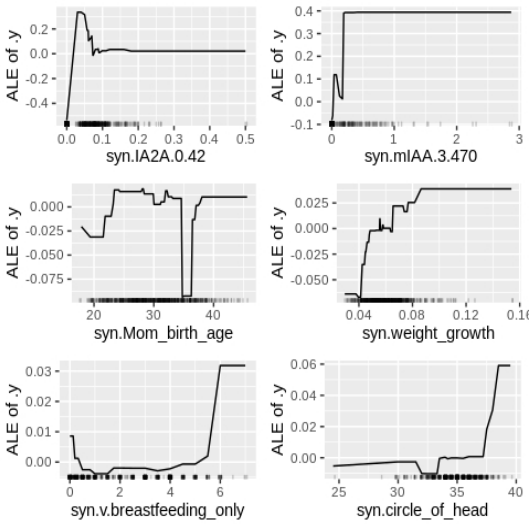


(b). Accumulated local effect plot

Figure 8. Original DIPP data

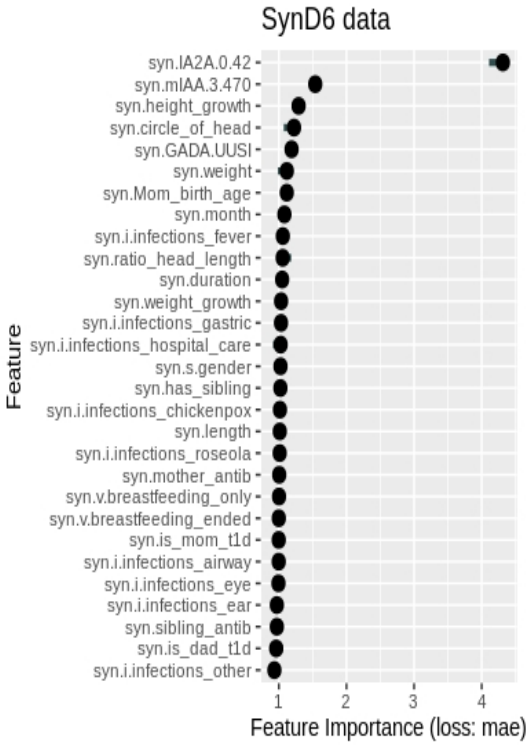


(a). Feature importance plot

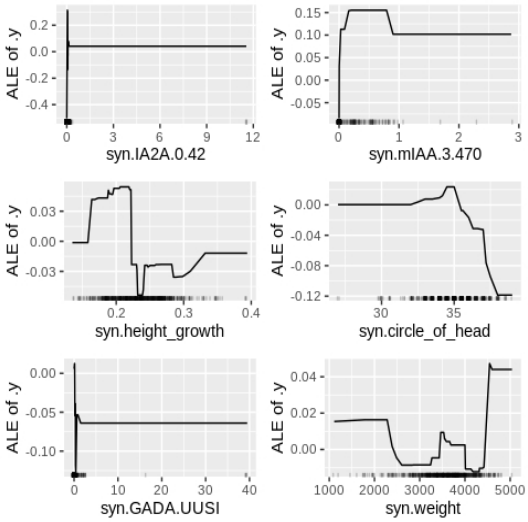


(b). Accumulated local effect plot

Figure 9. SynD1 data



(a). Feature importance plot



(b). Accumulated local effect plot

Figure 10. SynD6 data

4.1.2. General Utility

Following the performance evaluation of different methods of synthesis based on the specific utility of the original data set, In this section, we analyse and compare the statistical properties of the most reliable synthetic data set (*SynD1*) to statistical properties of the original data set. Comparative analyses include the calculation and measuring the changes in the Pearson correlation between variables, relative frequency distribution, data visualisation, and finally, the similarity between original and synthetic data set variables.

Pearson Correlation

The PPMCC matrix for original and *SynD1* data set can be seen in Figure 11. The lower triangle represents the Pearson correlation for the original data set, whereas the upper triangle represents the Pearson correlation for *SynD1* data set.

The Pearson correlation in Figure 11 for original and *SynD1* data set looks almost identical. However, the correlation between the variables `POS_antibodies` and IAA antibody is slightly stronger in the *SynD1* data set, ρ value is 0.13 in original and 0.37 in synthetic data set. Furthermore, the correlation between the variables `is_mom_t1d` and `mother_antib` suffered a slight decrease, with ρ value 0.3 in original and -0.01 in synthetic data set.

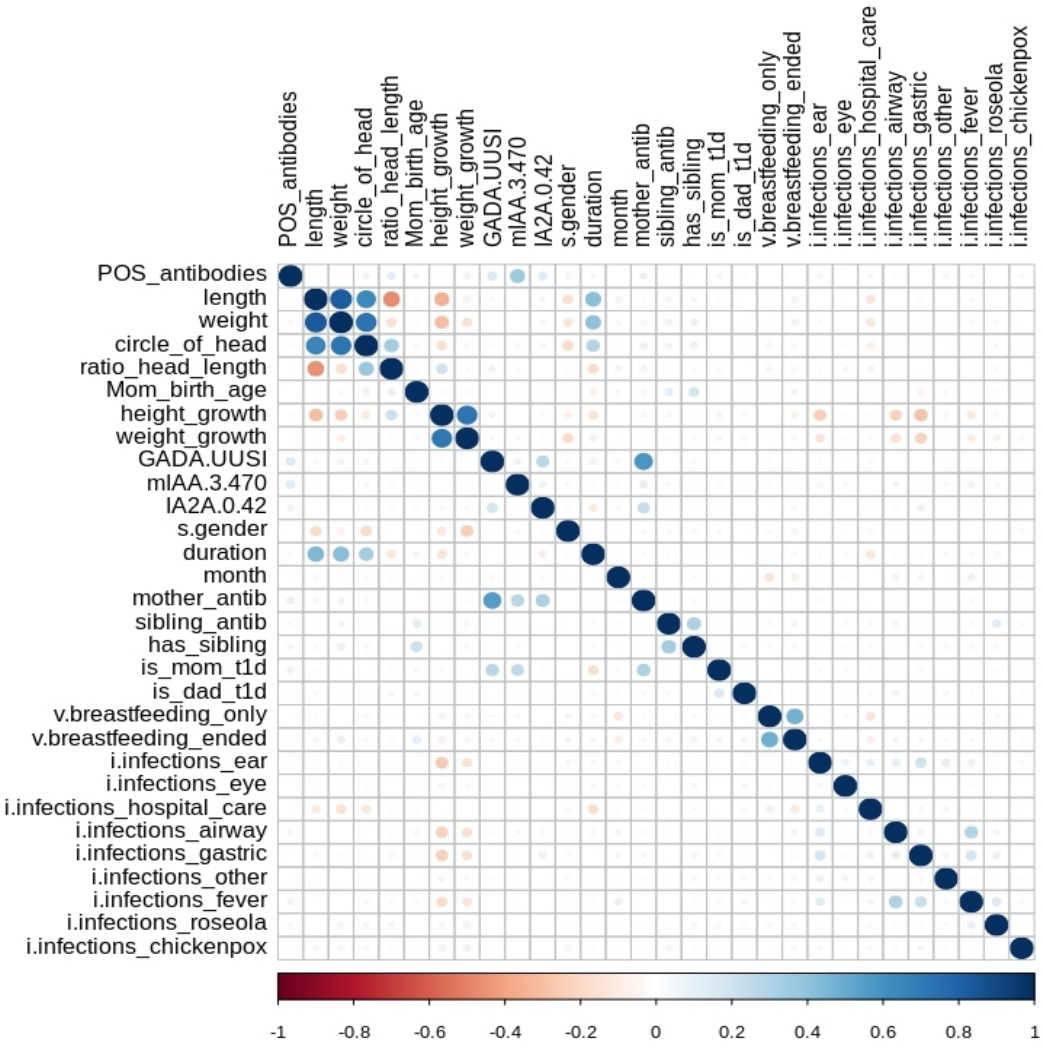


Figure 11. Pearson correlation for original data set in "lower" triangle and *SynD1* data set in "upper" triangle

Relative frequency distribution

The objective is to evaluate whether and to what degree the the data synthesis process preserves the probability distribution of the original data set. The relative frequency distributions of the original data set features in comparison with *SynD1* data set features were plotted to compare the likelihood of a specific result to occur in a given population.

Figures 12 and 13 shows the relative frequency distribution of a few variables from the original and *SynD1* data sets. The analysis revealed similar distributions between the original and synthetic data sets for every variable.

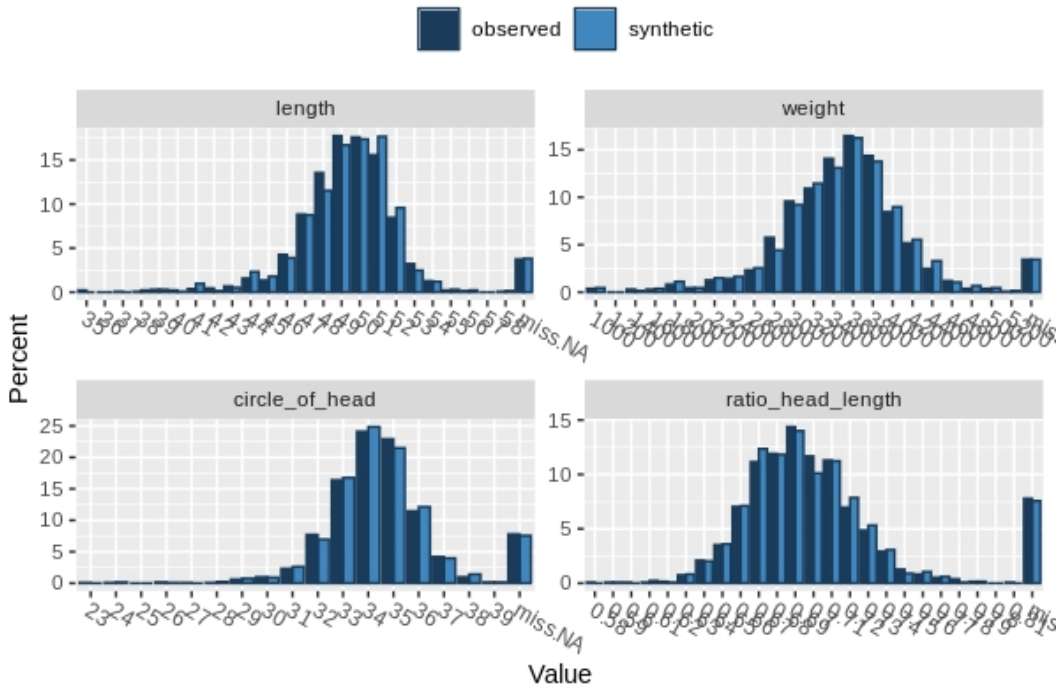


Figure 12. Relative frequency distribution of a few original (observed) and *SynD1* (synthetic) data set variables.

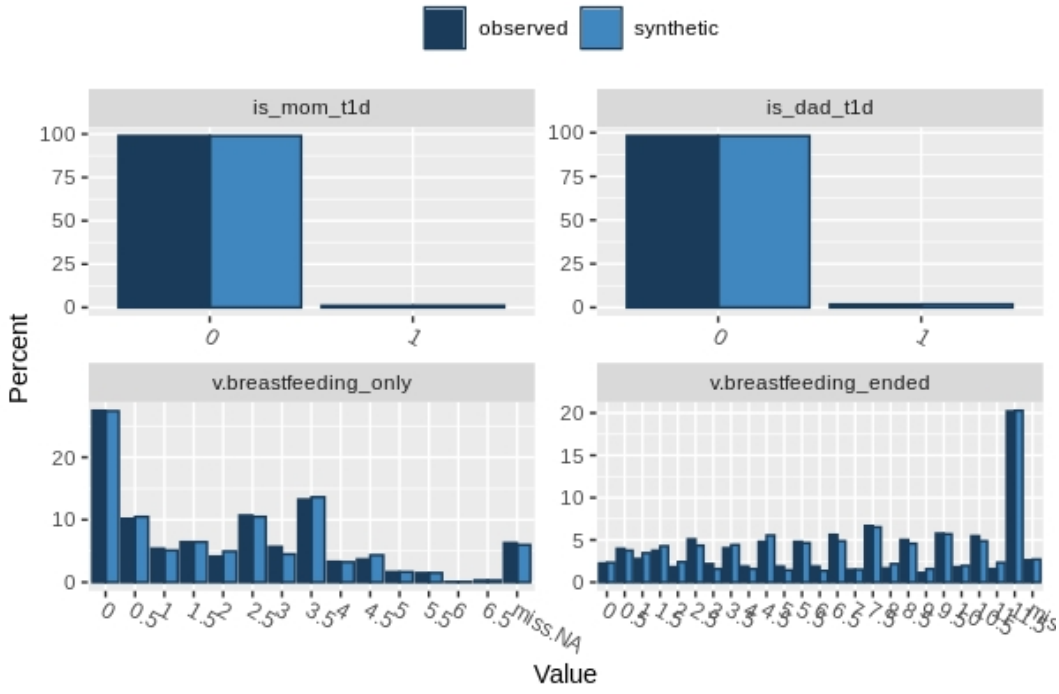


Figure 13. Relative frequency distribution of a few original (observed) and *SynD1* (synthetic) data set variables.

Uniform Manifold Approximation and Projection

UMAP can be used for dimension reduction of the data set before fitting the model; however, in our case, we are using UMAP tool to reduce the dimension to be able to visualise the data sets and evaluate the global and local structures within data before and after synthesis.

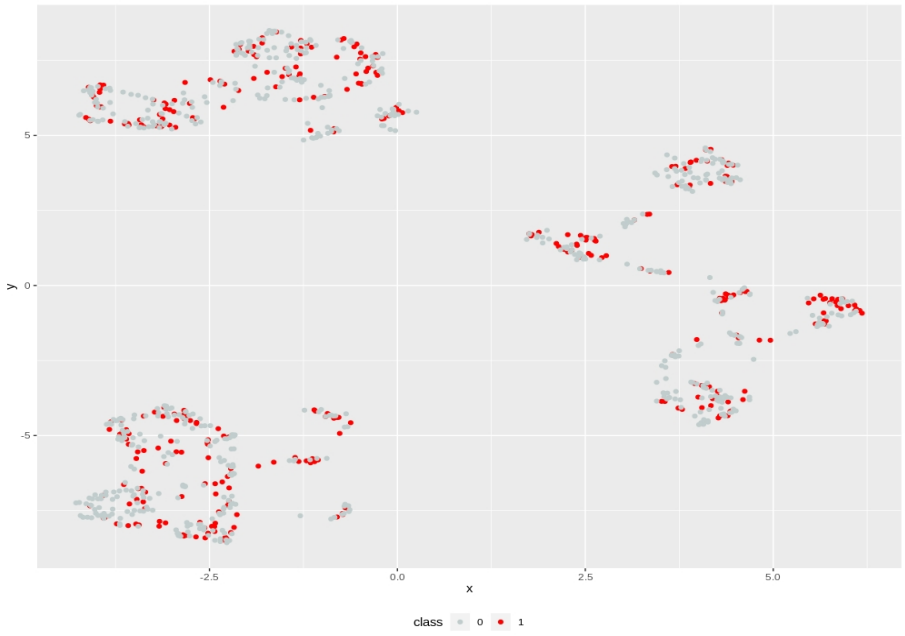


Figure 14. UMAP for original data set

Figure 15. UMAP for *SynD1* data set

Figures 14 and 15 shows the UMAP embedding for the original and *SynD1* data set respectively. In both figures, class 1 (positive) samples are represented by red colour dots and class 0 (negative) samples with grey colour dots. UMAP for the original data set (Figure 14) has approximately four clusters, whereas UMAP for *SynD1* data set (Figure 15) have two distinct clusters with other clusters more scattered as compared to the original data set.

Data Similarity

Data similarity of the continuous and discrete variables between original and *SynD1* data sets using Kolmogorov–Smirnov two-sample and Cucconi test can be seen in Table 12. From Table 12, all attributes have $kSp - value$ and Cucconi $p - value$ greater than α, which states that the analysis failed to reject the null hypothesis. In other words, the difference in the distribution of these variables are statistically non-significant.

Table 12. $KSp - value$ and Cucconi $p - value$ for matching continuous and discrete attributes between original and *SynD1* data sets.

Attribute	**KSp-value**	**Cucconi p-value**
length	0.7170990	0.603
weight	0.7924978	0.403
circle_of_head	1.0000000	0.914
ratio_head_length	0.9937073	0.495
Mom_birth_age	0.8930451	0.437
height_growth	0.9438003	0.629
weight_growth	0.7464065	0.472
GADA.UUSI	0.8380866	0.784
mIAA.3.470	0.5239224	0.965
IA2A.0.42	0.8097315	0.383
month	0.4346488	0.167
v.breastfeeding_only	0.9999954	0.946
v.breastfeeding_ended	0.9916316	0.981

4.1.3. Quality of Information Content

After analysing the impacts of data synthesis and usability of data from a data mining point of view, the concepts of information theory are used further to evaluate the level of distortion in a data set and quantify the information loss.

Entropy

Claude Shannon's entropy in bits was calculated for each variable in the data set. Figure 16 shows the entropy for each variable in bits before and after synthesis.

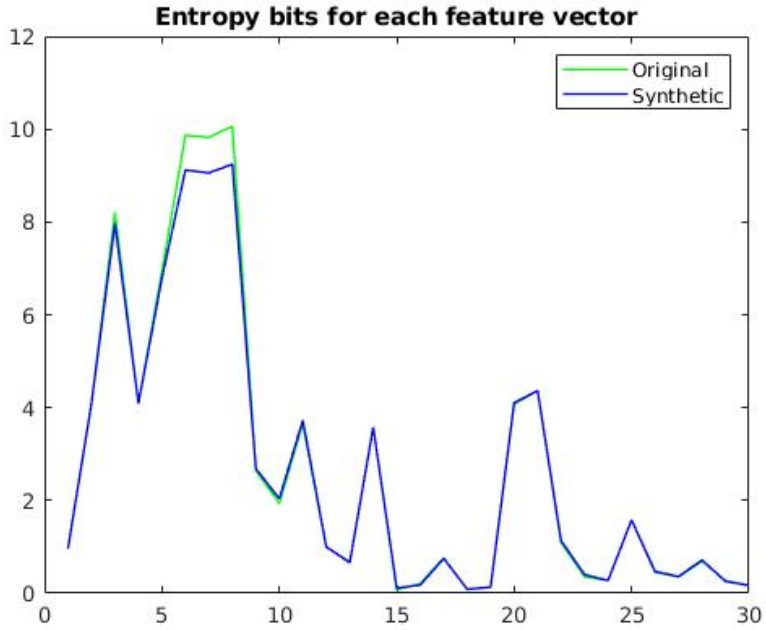


Figure 16. Entropy per bits for original and *SynD1* data variables.

From Figure 16, for almost all variables, except a few, entropy remains similar in both original and synthetic data set (*SynD1*). The number of variables corresponds to the order of variable in the Table 3. Variables such as the age of mother at the time of birth, growth rate of height and weight had a decrease in entropy by approximately one bit. A decrease in entropy bits states an increase in predictability of the values in the variables.

Mutual Information

Using a variation of kraskov's estimation method, MI for both original and synthetic data (*SynD1*) was calculated between the response variable `POS_antibodies` and rest 29 attributes. The number of neighbours was set to 6 for the k-NN algorithm, and the distance was calculated over a non-Euclidean space. Results revealed that MI remains the same in both data sets.

4.1.4. Outlines

Synthetic data set (*SynD1*) generated using `"cart"` method while not setting the argument `proper` to `TRUE`, performs adequately in all the analysis performed. The synthetic data showed no statistically significant difference from the original data set for performance on a machine learning model to predict the positivity of the autoantibodies later in life and dependence on variables. Furthermore, the synthetic data showed a similar bivariate correlation, relative and one-dimensional frequency distribution, and UMAP to the original data set. Additionally, the synthetic data revealed similar entropy bits for each variable and equal mutual information to the response variable.

4.2. HAR Using Smartphone Data Set

The pre-processed version of the HAR using smartphone data set was transformed into a data frame. With 562 columns, including the response variable and 10299 samples in total. The data set consists of multiple samples from the same subject for all six ADL. The data set was synthesised in multiple ways via `syn()` command from **Synthpop** package using `"cart"` method with a `seed` value for results replication. As mentioned earlier in subsection 3.1.2, the first variable to be synthesised in the data set is by default uses `"sample"` method as it does not have a predictor. In our case, the response variable is the first variable to be synthesised which is a class variable of six ADL converted into factors using `as.factor()` command in R, rest of the 561 attributes, are synthesised in the order they are found on HAR using smartphone data set repository under *features.text* file [16].

Synthesis of data set

The original data set was synthesised numerous times using `"cart"` method. Table 13 lists all denoted names of the synthetic data sets with the description of creation. The synthetic data sets *SynHAR1* and *SynHAR2* was randomly partitioned into two sets, where 70% of the subjects were selected for the training data and 30% the test data. For data set *SynHAR3*, 70% of original training data was synthesised and combined with the original training data, leaving with 140% data for training (70% original and 70% synthetic) and rest 30% original data was used for testing. Finally, for *SynHAR4*, leave-one-out cross-validation was implemented. The original data set was synthesised 30 time, each time original data from one subject was left out from synthesis process for testing.

Table 13. Denoted names of synthetic data sets with the description of creation

Synthetic data	Description
SynHAR1	Synthesis of full original data set
SynHAR2	Synthesis of train and test data separately
SynHAR3	Concatenate training set of SynHAR1 to training set of original data (increased training data)
SynHAR4	Synthesise data 30 times of 29 subject, leaving one subject out each time

4.2.1. Specific Utility

In this section, we examine whether the size of the data affects the performance of the primary tool of data synthesis during the data synthesis process. Moreover, we aim to demonstrate the applicability of one aspect of the secondary data analysis by testing the synthetic data fitted model with an original test set. Lastly, the concept of leave-one-out cross-validation while model testing was employed to illustrate the robustness of the synthesis tool.

Model performance with different data sets

After successfully synthesising the original data set multiple time, each data set from Table 13 along with the original data set was used for training and testing of three different machine learning models. Models used are NN with 18 neurons in the hidden layer, *k*-NN with 3 neighbours, and LDA. The number corresponding to the class are listed in Table 14.

Table 14. Class labels and corresponding activities.

Class	Activity
1	Walking
2	Walking Upstairs
3	Walking Downstairs
4	Sitting
5	Standing
6	Laying

Tables 15 and 16 list the performance of both original and *SynHAR1* data-fitted models, respectively. Table 17 shows the performance of different models when trained with synthetic data set (*SynHAR1*) and tested with the original data test set. Note that the training set of *SynHAR1* has samples from different subjects to that of the test set from original data for testing. Table 18 lists the performance of all three models when training and testing data was synthesised separately (*SynHAR2*). Table 19 lists the performance of all models when trained with a training set consists of both original and synthetic training set. Finally, Table 20 provides the performance summary of all previously mentioned experiments accompanying the average and maximum accuracy of *SynHAR4* using leave-one-out method on all 3 models.

From Table 20, the overall results indicate that the synthetic data sets give similar performance to that of the original data set. Additionally, performance of a test set of original data on *SynHAR1* data-fitted model is similar to performance on the original data-fitted model. However, a significant difference in the performance of models can be seen when the training and testing sets were synthesised separately (*SynHAR2*). Certainly from the performance of *SynHAR2*; we can say that the size of data affects the performance of the synthesis process due to the values are imputed by sampling from assumed distribution. In our case, the fitted parameters of the conditional distribution and the values from all previously synthesised columns were used. Even though all synthetic data sets were created using `"cart"` method, but their performance over different model diversifies. Additionally, results from *SynHAR4* data set reveals that the size of the data set for training and testing affects the performance of models.

Table 15. Performance of original data over different models

Model	Confusion Matrix								
			Target class						Accuracy
			1	2	3	4	5	6	
NN	Output class	1	**491**	9	4	0	0	0	97.4%
		2	0	**454**	19	4	0	0	95.2%
		3	5	8	**397**	0	0	0	96.8%
		4	0	0	0	**424**	11	0	97.5%
		5	0	0	0	63	**521**	13	87.3%
		6	0	0	0	0	0	**524**	100%
	Accuracy		99.0%	96.4%	94.5%	86.4%	97.9%	97.6%	**95.4%**
KNN			Target class						Accuracy
			1	2	3	4	5	6	
	Output class	1	**481**	36	49	0	0	0	85.0%
		2	4	**422**	42	3	0	0	89.6%
		3	11	13	**329**	0	0	0	93.2%
		4	0	0	0	**383**	57	1	86.8%
		5	0	0	0	105	**475**	1	81.8%
		6	0	0	0	0	0	**535**	100%
	Accuracy		97.0%	89.6%	78.3%	78.0%	89.3%	99.6%	**89.1%**
LDA			Target class						Accuracy
			1	2	3	4	5	6	
	Output class	1	**490**	11	1	0	0	0	97.6%
		2	6	**460**	15	1	0	0	95.4%
		3	0	0	**404**	0	0	0	100%
		4	0	0	0	**435**	22	0	95.2%
		5	0	0	0	55	**510**	0	90.3%
		6	0	0	0	0	0	**537**	100%
	Accuracy		98.8%	97.7%	96.2%	88.6%	95.9%	100%	**96.2%**

Table 16. Performance of Synthetic data (*SynHAR1*) over different models

Model	**Confusion Matrix**								
NN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**491**	13	6	0	0	0	96.3%
		2	9	**448**	31	2	0	0	91.4%
		3	1	9	**363**	0	0	0	97.3%
		4	0	1	0	**421**	75	0	84.7%
		5	0	0	0	55	**455**	0	89.2%
		6	0	0	0	2	0	**565**	99.6%
	Accuracy		98.0%	95.1%	90.8%	87.7%	85.8%	100%	**93.1%**
KNN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**478**	44	58	1	0	0	82.3%
		2	23	**423**	84	2	0	0	79.5%
		3	0	4	**258**	0	0	0	98.5%
		4	0	0	0	**349**	115	0	75.2%
		5	0	0	0	127	**415**	1	76.4%
		6	0	0	0	1	0	**564**	99.8%
	Accuracy		95.4%	89.8%	64.5%	72.7%	78.3%	99.8%	**84.4%**
LDA			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**490**	15	5	0	0	0	96.1%
		2	9	**449**	51	0	0	0	88.2%
		3	2	7	**344**	0	0	0	97.5%
		4	0	0	0	**415**	66	0	86.3%
		5	0	0	0	63	**464**	0	88.0%
		6	0	0	0	2	0	**565**	99.6%
	Accuracy		97.8%	95.3%	86.0%	86.5%	87.5%	100%	**92.5%**

Table 17. Performance of different models when trained with synthetic data training set (*SynHAR1*) and tested with original data test set.

Model	**Confusion Matrix**								
NN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**493**	34	5	0	0	0	92.7%
		2	2	**427**	29	0	0	0	93.2%
		3	1	10	**386**	0	0	0	97.2%
		4	0	0	0	**445**	44	1	90.8%
		5	0	0	0	46	**488**	0	91.4%
		6	0	0	0	0	0	**536**	100%
	Accuracy		99.4%	90.7%	91.9%	90.6%	91.7%	99.8%	**94.2%**
KNN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**488**	32	58	0	0	0	84.4%
		2	8	**435**	43	0	0	0	89.5%
		3	0	4	**319**	0	0	0	98.8%
		4	0	0	0	**374**	85	0	81.5%
		5	0	0	0	117	**447**	0	79.3%
		6	0	0	0	0	0	**537**	100%
	Accuracy		98.4%	92.4%	76.0%	76.2%	84.0%	100%	**88.2%**
LDA			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**493**	10	5	0	0	0	97.0%
		2	3	**460**	31	0	0	0	93.1%
		3	0	1	**384**	0	0	0	99.7%
		4	0	0	0	**453**	49	0	90.2%
		5	0	0	0	38	**483**	0	92.7%
		6	0	0	0	0	0	**537**	100%
	Accuracy		99.4%	97.7%	91.4%	92.3%	90.8%	100%	**95.4%**

Table 18. Performance of different models when training and testing data sets were separately synthesised (*SynHAR2*).

Model	**Confusion Matrix**								
NN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**429**	84	29	0	0	0	79.2%
		2	11	**384**	40	0	0	0	88.3%
		3	4	42	**352**	0	0	0	88.4%
		4	0	3	0	**382**	63	0	85.3%
		5	0	0	0	92	**462**	7	82.4%
		6	0	0	0	0	0	**563**	100%
	Accuracy		96.6%	74.9%	83.6%	80.6%	88.0%	98.8%	**87.3%**
KNN			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**418**	128	90	0	1	0	65.7%
		2	21	**369**	69	3	0	0	79.9%
		3	4	16	**262**	0	0	0	92.9%
		4	0	0	0	**262**	57	4	81.1%
		5	0	0	0	209	**467**	1	69.0%
		6	0	0	0	0	0	**565**	100%
	Accuracy		94.4%	71.9%	62.2%	55.3%	89.0%	99.1%	**79.5%**
LDA			**Target class**						**Accuracy**
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**424**	68	25	0	0	0	82.0%
		2	18	**426**	59	0	0	0	84.7%
		3	2	19	**337**	0	0	0	94.1%
		4	0	0	0	**352**	39	0	90.0%
		5	0	0	0	122	**486**	0	79.9%
		6	0	0	0	0	0	**570**	100%
	Accuracy		95.5%	83.0%	80.0%	74.3%	92.6%	100%	**88.1%**

Table 19. Performance of different models when training data is increased by adding a synthetic training set (*SynHAR3*).

Model	Confusion Matrix								
NN			Target class						Accuracy
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**490**	43	6	0	0	0	90.9%
		2	5	**411**	22	1	0	0	93.6%
		3	1	16	**392**	0	0	0	95.8%
		4	0	1	0	**441**	40	0	91.5%
		5	0	0	0	48	**492**	1	90.9%
		6	0	0	0	1	0	**536**	99.8%
	Accuracy		98.8%	87.3%	93.3%	89.8%	92.5%	99.8%	**93.7%**
KNN			Target class						Accuracy
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**478**	41	55	0	1	0	83.1%
		2	7	**422**	45	3	0	0	88.5%
		3	11	8	**320**	0	0	0	94.4%
		4	0	0	0	**389**	55	1	87.4%
		5	0	0	0	99	**476**	0	82.8%
		6	0	0	0	0	0	**536**	100%
	Accuracy		96.4%	89.6%	76.2%	79.2%	89.5%	99.8%	**88.9%**
LDA			Target class						Accuracy
			1	**2**	**3**	**4**	**5**	**6**	
	Output class	**1**	**489**	20	5	0	0	0	91.1%
		2	7	**451**	18	1	0	0	94.5%
		3	0	0	**397**	0	0	0	100%
		4	0	0	0	**437**	30	0	93.6%
		5	0	0	0	53	**502**	0	90.5%
		6	0	0	0	0	0	**537**	100%
	Accuracy		98.6%	95.8%	94.5%	89.0%	94.4%	100%	**95.5%**

Table 20. Accuracies of all data sets over different models including the accuracy of *SynHAR4* with leave-one-out method.

Data set		Model			
Training data	**Testing data**	**NN**	**KNN**	**LDA**	
Original	**Original**	95,3851	89,0736	96,2335	
SynHAR1	**SynHAR1**	93,0777	84,3909	92,5348	
SynHAR1	**Original**	94,1639	88,2253	95,3512	
SynHAR2	**SynHAR2**	87,2752	79,5385	88,0556	
SynHAR3	**SynHAR3**	93,7224	88,9379	95,4530	
SynHAR4	**Original**	92,6455	85,7729	93,5556	**Average**
	Original	99,4681	94,4149	99,6885	**Maximum**

4.2.2. General Utility

This section presents the analysis of the general utility of the synthetic and the original data set. The general utility measures difference in the statistical properties of a data set before and after the data synthesis process. The comparative measures used are Pearson correlation, relative frequency distribution, and data visualisation. For forthcoming measures, alone *SynHAR1* data set is used, representing the synthetic data set.

Pearson correlation

The PPMCC matrix of 20 out of 562 variables from the original and synthetic data set can be seen in Figure 17. The lower triangle represents the Pearson correlation for the original data set, whereas the upper triangle represents the Pearson correlation for the synthetic data set.

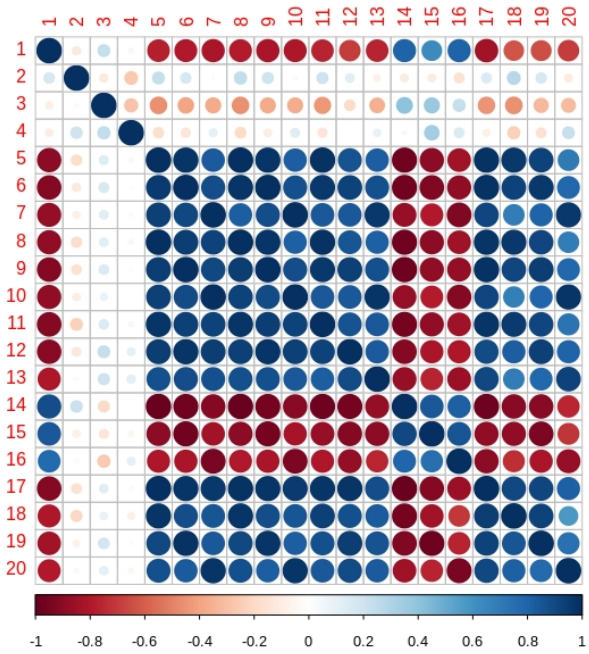


Figure 17. Pearson correlation for 20 variable from original and synthetic data set in "lower" triangle and "upper" triangle, respectively

From the correlation matrix (Figure 17), both original and synthetic data set has strong correlations between variables. However, correlations between variable number 3 and the rest of the variables are stronger in the synthetic data set in comparison to the original data set.

Relative frequency distribution

The relative frequency distributions for a few variables of the original and synthetic data set can be seen in Figures 18 and 19, respectively. The analysis reported similar distributions between the original and synthetic data sets.

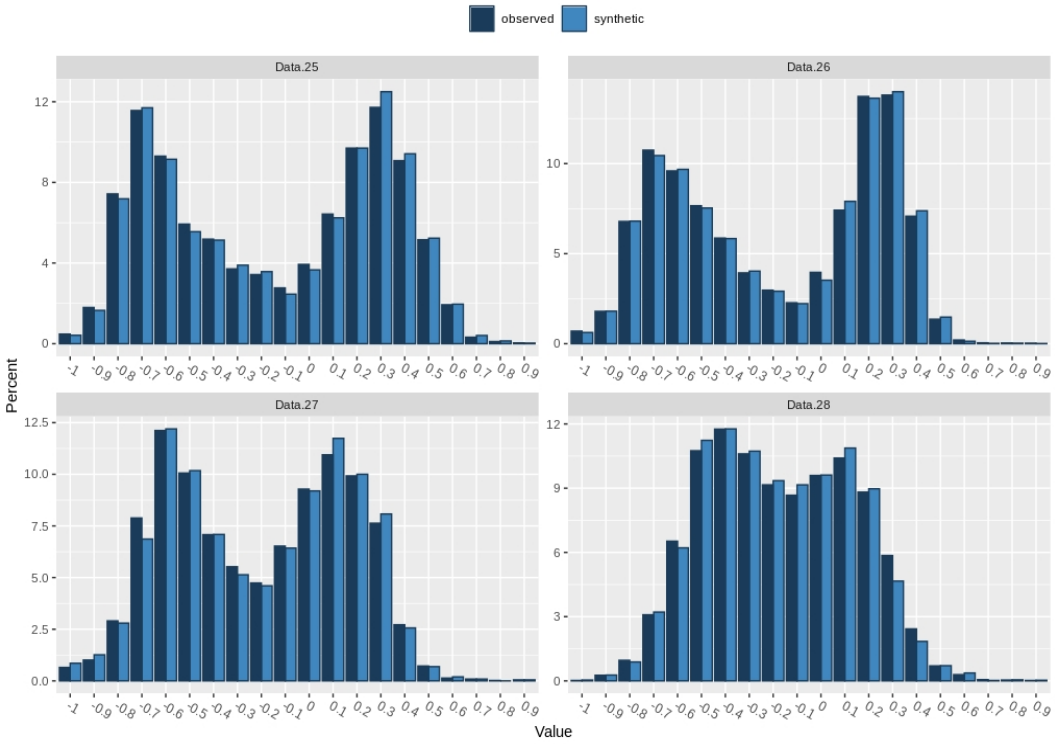


Figure 18. Relative frequency distribution of a few original (observed) and synthetic data set variables.

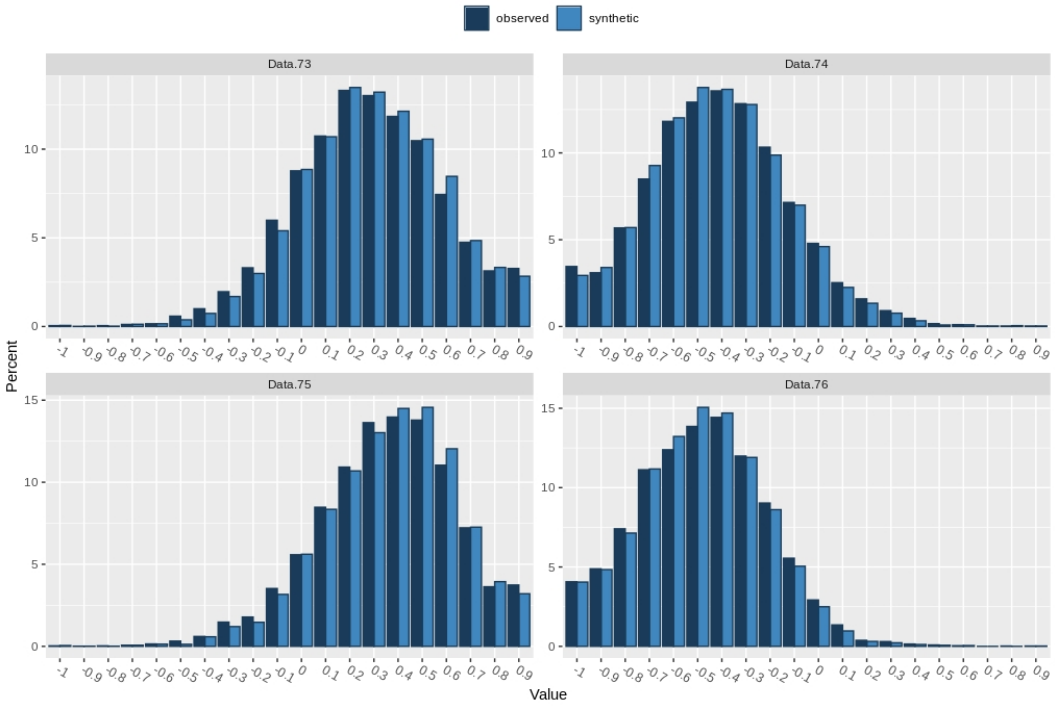


Figure 19. Relative frequency distribution of a few original (observed) and synthetic data set variables.

Uniform Manifold Approximation and Projection

The UMAP embedding for original and synthetic data are shown in Figures 20 and 21, respectively. In UMAP for both original and synthetic data sets, samples belonging to class 6 forms an exclusive cluster, and samples from classes 4 and 5 builds entirety connected cluster. However, samples from classes 1, 2, and 3 occur scattered and scarcely make one cluster in the original data set, but forms a rather tight single cluster in synthetic data set; suggesting that synthesis of data altered the local structures within data set. Additionally, the global structures are affected as well; since the distance between classes have shortened for some samples.

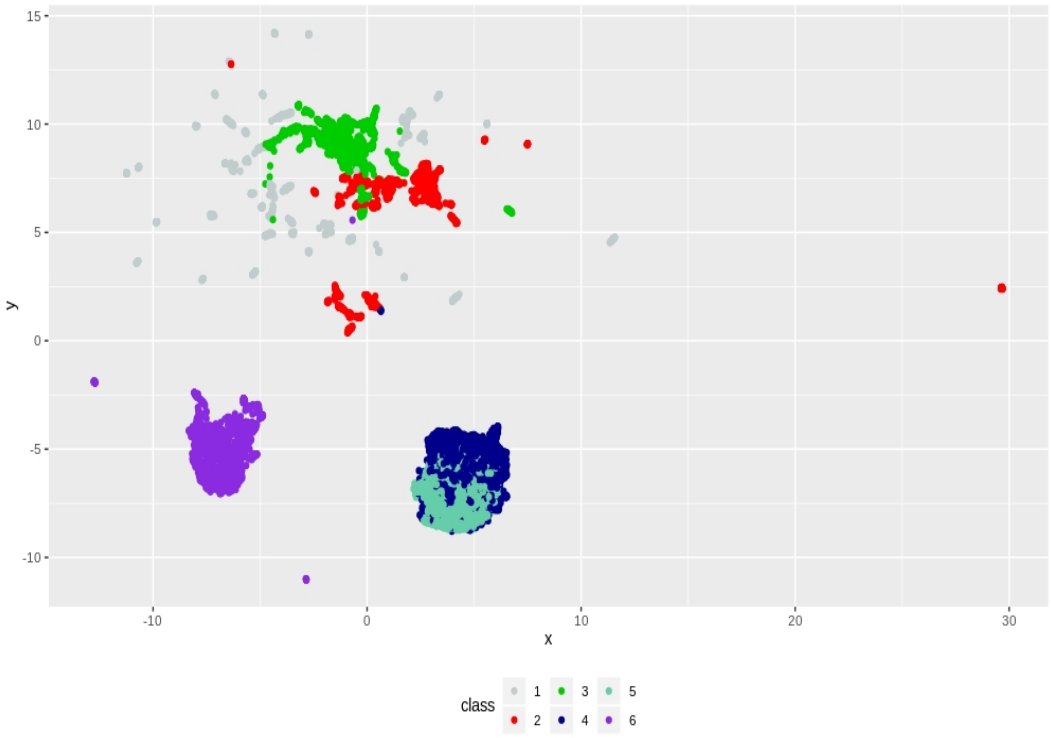


Figure 20. UMAP for original data set

4.2.3. Quality of Information Content

Entropy

Entropy in bits, calculated for all variables concerning both original and synthetic data set can be seen in Figure 22. Each variable in synthetic data set suffered a drop in entropy by approximately one bit, suggesting a probability of data compression, which can lead to an increase in predictability of values for each data variable.

Mutual Information

MI between all feature vectors and the response variable for both original and synthetic data set was calculated using a variation of Kraskov's estimation method. The distance between samples was calculated using a k-NN algorithm with 3 neighbours over a

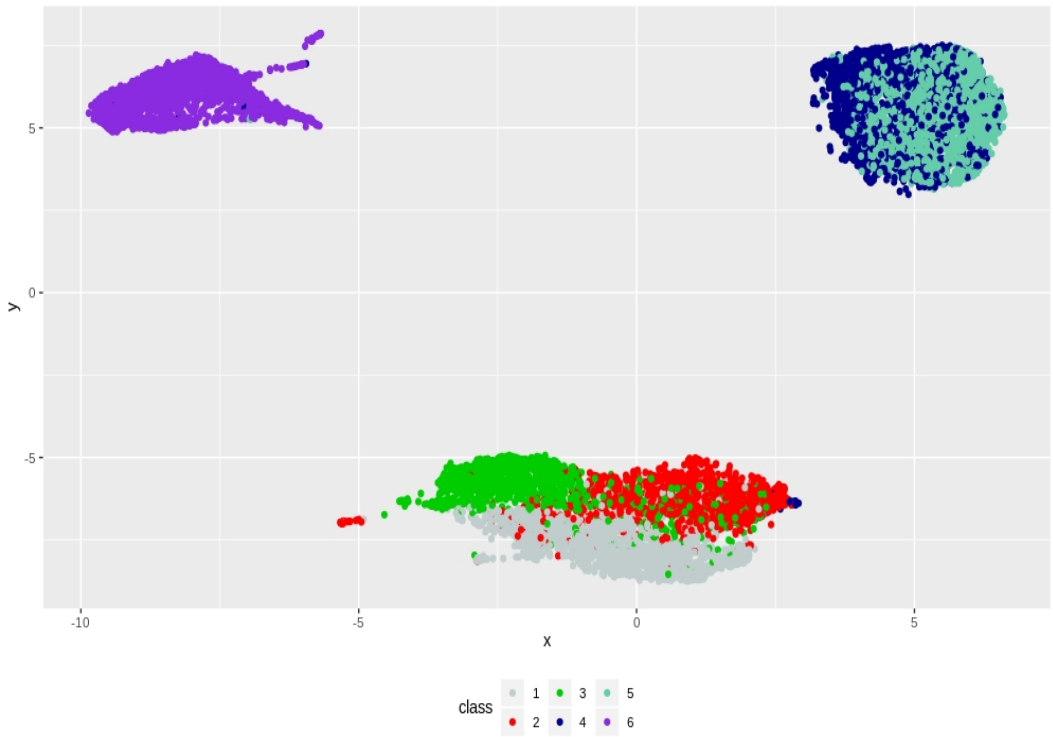


Figure 21. UMAP for synthetic data set

non-Euclidean space. The analysis acknowledges no variation in MI in the synthetic data set as compared to the original data set.

4.2.4. Outlines

The synthetic data produced using `"cart"` method performs reasonably well on all three machine learning models. Furthermore, the original data set seemed to give a similar performance on the synthetic data fitted model. However, overall utilities of synthetic data suffered a drop in performance according to the analyses performed—additionally, the size of the data to be synthesised effects the performance. Because the data was divided into two sets for training and testing and was synthesised separately, the performance of the fitted models dropped drastically. Overall, the bivariate correlations between the variables in both synthetic and original data remained presumably maintained except for one variable. The relative frequency distribution analysis revealed a similar distribution for all variables. However, the UMAP revealed a notable change in the global and local structures within the data set. Finally, all the variables suffered a decline in entropy by 1-bit. Nevertheless, Mutual information between class variable and features remained unchanged.

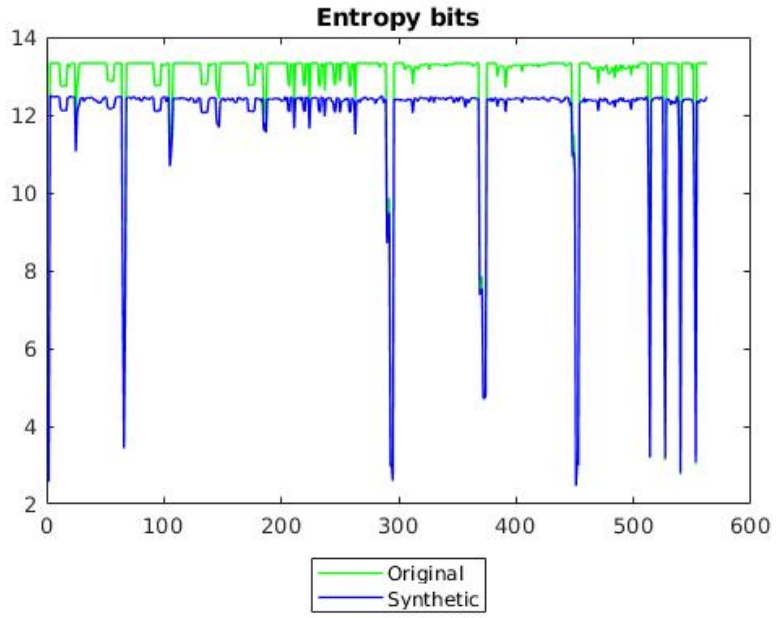


Figure 22. Entropy per bits for original and synthetic data variables.

5. DISCUSSION

The objective of the thesis was to assess the performance of a tool for data synthesis, termed **Synthpop** by estimating the impacts of the data synthesis process. In order to accomplish the aforementioned goal, the characteristics of the primary tool of data synthesis were described and utilised to generate synthetic data sets. Next, various data standards were established based on the general and specific utility, and quality of information contained in the original data set. The general utilities are the statistical properties of a data set, whereas the specific utilities are the performances of a data-fitted model. Lastly, the quality of information content is the entropy and MI within a data set. After successfully establishing the data standards, the impacts of the data synthesis process were measured from the differences in the data set before and after synthesis, based on the established standards. A synthetic data set is considered adequate when two requirements are met: 1) the difference or change in the synthetic data utilities as compared to the original data are statistically non-significant and 2) the quality of the information content in synthetic data is similar to that of the original data set. The study utilised two distinct data sets to assess the comprehensiveness of the tool for data synthesis.

5.1. Principal Discoveries

5.1.1. DIPP Data Set

It is natural to see a healthcare database suffering from imbalanced classes. Especially in real-world data, this is often expected as the data is not gathered in an experimental setting such as randomised controlled trial, instead collected in real-world settings, such as from patient surveys, clinical trials, and observational cohort studies. It is necessary to note that such characteristics affect the performance of machine learning algorithms [56, 69]. In our case, it is not the scope of the study to investigate this reasoning and improve the performance; however, it is essential as it has also affected the data synthesis process. The tool imputes the value for synthetic variable from fitted parameters of synthesising models, and imbalanced classes played a significant role in most of the synthesising methods.

The DIPP data set was pre-processed and mostly aggregated from a longitudinal database. When such data is generated, it is expected that the data set will suffer from imbalanced classes. Such characteristic play a significant role in data analysis; in our case, the effects can be seen in model training. Most of the synthetic data-fitted models, including the original data-fitted model performed reasonably well in the prediction of negative and positive cases despite having a high number of negative samples. However, one synthetic data set (*SynD5*) significantly favoured the negative samples more than any other data set. The *SynD5* was the only synthetic data set which was produced using parametric methods fitting the type of data variables. This analysis suggests that the during data synthesis, model fitting parameters of the synthesising method might have suffered overfitting, and synthetic data values were imputed to favour negative classes. Even though the significance test of the accuracies of the *SynD5* data-fitted model reports no statistically significant difference to the original

data-fitted model, when other evaluation parameters were considered, the *SynD5* revealed various shortcomings favouring the previous finding. These interpretations underline the importance of the other evaluation parameters while determining a model's performance.

On the other hand, non-parametric methods have generated synthetic data which not only fell behind but also exceeded the performance of the original data-fitted model. For example, *SynD2*, *SynD3* and *SynD4* synthetic data-fitted model show the accuracies of 86.0%, 90.0% and 93.0%, respectively. The difference in the performance could be again due to overfitting or underfitting of model fitting parameters of synthesising methods. However, it could also be induced by a variation in bivariate correlations between variables during data synthesis. All of these data sets (*SynD2, SynD3, and SynD4*) also showed a statistically significant difference in model accuracies with the original data set. From these analyses, we can say that the specific utility of synthetic data is highly dependent on the method of synthesis. The issues could be resolved by thoughtfully selecting a different method for data synthesis according to the type of data. Moreover, we can also control the model fitting parameters of the synthesising method by controlling the `predictor.matrix` to avoid overfitting and underfitting of the synthesising model.

Despite its weaknesses, the tool exceeded the expectations when the default method of synthesis `"cart"` was used, which can handle any data type. Two synthetic data sets were generated using `"cart"` method: *SynD1* and *SynD6*. The only difference was that *SynD6* data was generated while setting the argument `proper` to `TRUE` for proper synthesis. Repeatedly, the *SynD6* data-fitted model showed signs of overfitted parameters of synthesising model during data synthesis—however, the *SynD1* data-fitted model outperformed in all analyses. The synthetic data set showed no signs of variation in data utility. Synthetic data set *SynD1* succeeded at all performed tests with the statistically non-significant difference from the original data set; this is the only synthetic data set which leads to failing to reject the null hypothesis. Additionally, the quality of the information content was also well preserved for 27 out of 30 variables. For the rest of the three variables, *SynD1* suffered a decrease in entropy only by 1-bit. Conclusively, these analyses suggest that the `"cart"` method not only preserved the utilities but also preserved the complexity of the DIPP data set according to the data standard established in this study; exhibiting that the tool certainly accomplished its intended goal.

5.1.2. HAR Data Set

The whole HAR data set has an undeniably strong correlation between features, since all of them are generated utilising two initial raw signals obtained from an accelerometer and gyroscope of mobile devices. Such a data set could be challenging to replicate fully, as similar variables are used to derive different features, a strong dependence between the features is expected. Even though **Synthpop** imputes values from the assumed distribution, conditional on all previously observed variables, the fitted parameters of the conditional distribution, and all previously synthesised variables, the effects of the size and complexity of data can be seen in the performance

of data synthesis tool. Nevertheless, the tool performed adequately considering that advanced and rigorous experiments were employed for the HAR data set.

Comprehending from all the previous conclusions from the DIPP data set, we generated various synthetic HAR data sets practising `"cart"` method with diverse techniques. Due to the size of data and high complexity of features, the performance of the data synthesis tool suffered. The entropy bits for all features in synthetic data had a reduction by 1-bit as opposed to the entropy of original data features. This implies that the synthesising method was not able to preserve the complexity of the data and further shows the signs of data compression. The analyses also revealed an accumulated drop in the performance of all synthetic data-fitted models as compared to the original data-fitted model. The reduction in performance could again be due to a decline in complexity occurred and data compression can be interpreted as lossy data compression from entropy analysis. The formerly mentioned finding can be further backed up from the performance outcomes of *SynHAR3* data-fitted model. Since the training data had both synthetic and original training data sets, the model's performance was almost similar to the original data-fitted model, which signifies the loss of information in rest of the synthetic data sets. Nevertheless, the value for mutual information between the data sets remained unchanged.

As anticipated, the distribution of the data played a significant role as well. When we synthesised the training and testing data separately (*SynHAR2* data set), needless to say, the performance of the synthetic data over machine learning models dropped drastically. Subsequent drop in the performance could be due to a change in distribution within the data set, caused by dividing the data into two separate sets. This finding suggests that whenever more data is collected, the original data should be synthesised again as a whole to achieve the best results from the synthesising tool. We also intended to demonstrate the robustness of the synthesis tool by appropriating the concept of leave-one-out cross-validation for more intensive model evaluations. The analysis revealed similar results. In terms of general utility, the *SynHAR1* data set held overall alike bivariate correlations and relative frequency distributions to the original data set. However, the Uniform Manifold Approximation and Projection of the HAR original and synthetic data set showed small indications of changes in the local and global structures of the data.

Nonetheless, the overall performance of the data synthesis tool was remarkable. The tool performed adequately on all performed tests. Furthermore, the performance of the synthetic data-fitted model was similar to the original data-fitted model when both models were tested with an original data test set, which shows excellent signs of secondary data analysis. We also conclude that **Synthpop** could perform even better for the HAR data set by applying different synthesising methods. Since our previous finding showed that the method of synthesis plays a significant role in the performance of the synthesis tool, choosing a method of synthesis specific to the data can improve the quality of the synthetic data. In this study, all synthetic HAR data sets were generated using `"cart"` method, since all previous analyses from DIPP data showed the exceptional performance of the synthesising method. However, different methods of synthesis could improve the performance of synthesising tool even further.

5.2. Synopsis and Future Work

The impediments in healthcare data mining and sharing most often relate to research participant's or patient's privacy, security, and the circumstance that researchers face of having to consider the trade-off between the risk of disclosure and the benefits of open data sets [10, 70, 71, 72]. Open healthcare data not only benefits extended scientific collaboration for innovative discoveries and validating previously defined hypotheses but more importantly, sharing healthcare data could save lives. Healthcare databases are demonstrating to play an indispensable part in controlling and preventing the spread of the novel coronavirus "COVID-19" (SARS-CoV-2) in a worldwide pandemic [73, 74, 75, 76]. In numerous situations, the survival of a database itself depends on the data holder's capability to provide data when needed, since not releasing such data at all may eventually diminish the need for it [11]. However, the process of sharing healthcare data needs careful measures as it could unfold severe consequences through the risk of disclosure and could harm not only the participants but also organisations or individuals involved in collecting and sharing data [77].

Current data sharing systems, including SQLShare [78] and DataHub [79], promote collaborative data analyses but fail to consolidate privacy-preserving prospects or means to manage sensitive data. **Synthpop** could amend this by producing a synthetic version of the original data set. Furthermore, the use of synthetic data for secondary data analysis will enhance the collaboration between data owners and external data scientists while maintaining the subject's privacy. However, the achievement of anonymity relies on the assumption that there are no matching samples between the original and synthetic data sets, also, there are no samples with extreme values which could serve as a unique identifier. Additionally, the utility of the data is highly dependent on the performance of the synthesising model, and the **Synthpop** package itself provides minimal tests for the synthetic data analysis. Comparing only the relative frequency distribution of two data sets for statistical analysis says a lot but from a rather vague perspective. Furthermore, the package provides the comparison of the data-fitted models but only with linear machine learning techniques which is again somewhat limited.

However, as demonstrated in this study, a user could utilise different tools to measure the utility of the data or consolidate further questioning if desired. Subsequently studying and assessing **Synthpop** by measuring the impacts of the data synthesis process, we conclude that the tool performs competently in the current setting. Future researchers could consider testing the performance of **Synthpop** by synthesising the HAR data set using different methods of synthesis. Furthermore, implementing a more sophisticated way to read entropy bits and investigating the mutual information between pairs of variables in both original and synthetic data sets could highlight more in-depth impacts of the data synthesis process. Further analyses, including the involvement of different tools for data anonymisation such as differential privacy, can provide a comparative analysis from a different point of view. Subsequently, examining the performance of **Synthpop** on longitudinal data could provide a greater understanding of the comprehensiveness of the tool. Finally, more advanced analyses are required to question whether the assumed anonymity in the synthetic data set exists, and to what extent.

6. CONCLUSION

The thesis was inspired by the benefits of open healthcare databases and aimed to examine a unique solution to perpetual hindrances in data sharing caused by the risk of disclosure and shortcomings of current data anonymisation techniques. Therefore, in this thesis, the performance of a tool for data synthesis, termed **Synthpop**, was analysed by assessing the impacts of the data synthesis process. Impacts were measured based on the quantifiable changes in utilities and quality of information contained in the data set before and after synthesis. Two different types of data sets were used to evaluate the generalisability of the data synthesis tool. Our statistical analyses conclude that the tool is generalised in terms of applicability to different data types. Furthermore, synthetic data mimics the original data set while preserving all the statistical properties, machine learning capabilities, and quality of the information contained in the original data set with statistically non-significant differences. In conclusion, the tool succeeded at its intended purpose and can be used to generate synthetic data sets for data sharing purposes. However, the performance of the tool profoundly depends on the method of synthesis, i.e., carefully choosing a method of synthesis improves the performance of the tool.

Overall, **Synthpop** fulfils all the necessities towards data sharing and hence unfolds a wide range of opportunities in the research community, including easy data sharing, more significant collaborations, and information protection [70]. Considering the workflow of the study, we can also state that data collectors and authors will always be indulged, since the findings from the synthetic data need verification from the original data set. This dependency on the original data set for result verification embeds a limitation on the study because the synthetic data can only be used for secondary data analysis. If the original author can not be reached for result verification, the analyses may cease and result in an abandoned study.

7. REFERENCES

[1] Ganz F., Barnaghi P. & Carrez F. (2013) Information abstraction for heterogeneous real world internet data. IEEE Sensors Journal 13, pp. 3793–3805. DOI: 10.1109/JSEN.2013.2271562.

[2] Reinsel D., Gantz J. & Rydning J. (2018) The digitization of the world from edge to core. IDC White Paper URL: https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf, Accessed 12.06.20.

[3] Desjardins J. (2019) How much data is generated each day. Visual Capitalist. Recuperado el 14. URL: https://www.visualcapitalist.com/how-much-data-is-generated-each-day/, Accessed 15.11.19.

[4] Why is data important for your business. URL: https://www.grow.com/blog/data-important-business, Accessed 17.11.2019.

[5] Regalado A. (2013) The data made me do it. MIT Technology Review 116, pp. 63–64. URL:https://www.technologyreview.com/2013/05/03/16109/the-data-made-me-do-it/, Accessed 17.11.2019.

[6] Morey T., Forbath T. & Schoop A. (2015) Customer data: Designing for transparency and trust. Harvard Business Review 93, pp. 96–105.

[7] Rocher L., Hendrickx J.M. & De Montjoye Y.A. (2019) Estimating the success of re-identifications in incomplete datasets using generative models. Nature Communications 10, pp. 1–9. DOI: 10.1038/s41467-019-10933-3.

[8] Yale A., Dash S., Dutta R., Guyon I., Pavao A. & Bennett K.P. (2019) Privacy Preserving Synthetic Health Data. In: Proc. 2019 ESANN, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges (Belgium), Apr 2019, pp. 465–470.

[9] General Data Protection Regulation. URL: https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/sensitive-data/what-personal-data-considered-sensitive_en. Accessed 28.10.2019.

[10] Cios K.J. & Moore G.W. (2002) Uniqueness of medical data mining. Artificial Intelligence in Medicine 26, pp. 1–24.

[11] Sweeney L. (2002) k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10, pp. 557–570. DOI: 10.1142/S0218488502001648.

[12] Viceconti M., Hunter P. & Hose R. (2015) Big data, big knowledge: big data for personalized healthcare. IEEE Journal of Biomedical and Health Informatics 19, pp. 1209–1215. DOI: 10.1109/JBHI.2015.2406883.

[13] Lebied M., Misleading Statistics Examples – Discover The Potential For Misuse of Statistics & Data In The Digital Age. URL: https://www.datapine.com/blog/misleading-statistics-and-data/. Accessed 30.11.2019.

[14] Badr W., Why Feature Correlation Matters .... A Lot! URL: https://towardsdatascience.com/why-feature-correlation-matters-a-lot-847e8ba439c4. Accessed 29.11.2019.

[15] Finnish Type 1 Diabetes Prediction and Prevention. URL: http://dipp.fi. Accessed 21.01.2020.

[16] Anguita D., Ghio A., Oneto L., Parra X. & Reyes-Ortiz J.L. (2013) A public domain dataset for human activity recognition using smartphones. In: Proc. 2013 ESANN, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges (Belgium), pp. 437–442.

[17] Van Ginneken A.M. (2002) The computerized patient record: balancing effort and benefit. International Journal of Medical Informatics 65, pp. 97–119.

[18] Huston P., Edge V. & Bernier E. (2019) Open science/open data: Reaping the benefits of open data in public health. Canada Communicable Disease Report 45, p. 252. DOI: 10.14745/ccdr.v45i10a01.

[19] Florez J.C. (2016) Precision medicine in diabetes: is it time? Diabetes Care 39, pp. 1085–1088. DOI: 10.2337/dc16-0586.

[20] Ross C. & Swetlitz I. (2017) Ibm pitched its watson supercomputer as a revolution in cancer care. it's nowhere close. Stat https://www.preventcancer.org/wp-content/uploads/2018/06/IBM_pitched_Watson_as_a_revolution_in_cancer_care.pdf, Accessed 22.01.2020.

[21] Ngufor C., Van Houten H., Caffo B.S., Shah N.D. & McCoy R.G. (2019) Mixed effect machine learning: a framework for predicting longitudinal change in hemoglobin a1c. Journal of Biomedical Informatics 89, pp. 56–67. DOI: 10.1016/j.jbi.2018.09.001.

[22] Greely H.T. (2007) The Uneasy Ethical and Legal Underpinnings of Large-Scale Genomic Biobanks. Annual Review of Genomics and Human Genetics. DOI: 10.1146/annurev.genom.7.080505.115721.

[23] Gola P. & Schomerus R. (2010) Bdsg kommentar. Auflage CH Beck, München , p. 47.

[24] FERPA (1997), Family educational rights and privacy act (usc 1232-34 cfr part 99). US Department of Education, Washington, DC. URL: https://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html Accessed 16.02.20.

[25] Fellegi I.P. (1972) On the question of statistical confidentiality. Journal of the American Statistical Association 67, pp. 7–18. DOI: 10.1080/01621459.1972.10481199.

[26] Denning D.E. (1980) Secure statistical databases with random sample queries. ACM Transactions on Database Systems (TODS) 5, p. 295. DOI: 10.1145/320613.320616.

[27] Purdam K. & Elliot M. (2007) A case study of the impact of statistical disclosure control on data quality in the individual uk samples of anonymised records. Environment and Planning A 39, pp. 1101–1118. DOI: 10.1068/a38335.

[28] Samarati P. & Sweeney L. (1998) Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report SRI-CSL-98-04, SRI Computer Science Laboratory, Palo Alto, CA.

[29] Machanavajjhala A., Kifer D., Gehrke J. & Venkitasubramaniam M. (2007) l-diversity: Privacy beyond k-anonymity. Association for Computing Machinery, Transactions on Knowledge Discovery from Data, pp. 24 DOI: 10.1145/1217299.1217302.

[30] Li N., Li T. & Venkatasubramanian S. (2007) t-closeness: Privacy beyond k-anonymity and l-diversity. In: Proc. 2007 IEEE, 23rd International Conference on Data Engineering, pp. 106–115. DOI: 10.1109/ICDE.2007.367856.

[31] Dwork C., McSherry F., Nissim K. & Smith A. (2006) Calibrating noise to sensitivity in private data analysis. In: Proc. 2006 Springer, Theory of Cryptography Conference, pp. 265–284. DOI: 10.1007/11681878_14.

[32] Erlingsson Ú., Pihur V. & Korolova A. (2014) Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proc. 2014 ACM, Special Interest Group on Security, Audit and Control (SIGSAC) Conference on Computer and Communications Security, pp. 1054–1067. DOI: 10.1145/2660267.2660348.

[33] Info Apple P. (2016), Apple previews ios 10, the biggest ios release ever. URL: https://www.apple.com/newsroom/2016/06/apple-previews-ios-10-biggest-ios-release-ever/. Accessed 15.04.2020.

[34] Ohm P. (2009) Broken promises of privacy: Responding to the surprising failure of anonymization. UCLA Law Review 57, p. 1701.

[35] Culnane C., Rubinstein B.I. & Teague V. (2017) Health data in an open world. arXiv preprint arXiv:1712.05627v1 .

[36] General Data Protection Regulation. URL: https://gdpr.eu/tag/chapter-3/. Accessed 14.01.2020.

[37] Nowok B., Raab G.M., Dibben C. et al. (2016) Synthpop: Bespoke creation of synthetic data in r. Journal of Statistical Software 74, pp. 1–26. DOI: 10.18637/jss.v074.i11.

[38] Arslan R.C., Schilling K.M., Gerlach T.M. & Penke L. (2018) Using 26,000 diary entries to show ovulatory changes in sexual desire and behavior. Journal of Personality and Social Psychology DOI: 10.1037/pspp0000208.

[39] Breiman L. (2001) Random forests. Machine Learning 45, pp. 5–32.

[40] Snoke J., Raab G., Nowok B., Dibben C. & Slavkovic A. (2018) General and specific utility measures for synthetic data. Journal of the Royal Statistical Society. Series A: Statistics in Society 181, pp. 663–688. DOI: 10.1111/rssa.12358.

[41] Manikandan S. (2011) Frequency distribution. Journal of Pharmacology & Pharmacotherapeutics 2, p. 54. DOI: 10.4103/0976-500X.77120.

[42] Kolmogorov A. (1933) Sulla deterrninazione empirica di una legge di distribuzione. Giornale dell'Istituto Italiano degli Attuari 4, pp. 89–91.

[43] Maaten L.v.d. & Hinton G. (2008) Visualizing data using t-sne. Journal of Machine Learning Research 9, pp. 2579–2605.

[44] McInnes L., Healy J. & Melville J. (2018) Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426v2 .

[45] Schapire R.E. (1990) The strength of weak learnability. Machine Learning 5, pp. 197–227. DOI: 10.1007/BF00116037.

[46] Freund Y. (1995) Boosting a weak learning algorithm by majority. Information and Computation 121, pp. 256–285. DOI: 10.1006/inco.1995.1136.

[47] Freund Y., Schapire R.E. et al. (1996) Experiments with a new boosting algorithm. In: 13th International Conference proceedings, Machine Learning, pp. 148–156.

[48] Friedman J., Hastie T., Tibshirani R. et al. (2000) Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). The Annals of Statistics, 28 , pp. 337–407.

[49] Friedman J.H. (2001) Greedy function approximation: a gradient boosting machine. Annals of Statistics , pp. 1189–1232.

[50] Candel A. & Malohlava M. (2020) Gradient Boosted Models. URL: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/GBMBooklet.pdf, Accessed 21.12.19.

[51] Friedman J., Hastie T. & Tibshirani R. (2001) The elements of statistical learning, 1. Springer Series in Statistics New York.

[52] Matlab (R2019b) Deep Learning Toolbox. The MathWorks, Inc., Natick, Massachusetts, United State. URL: https://se.mathworks.com/products/statistics.html, Accessed 02.02.20.

[53] Matlab (R2019b) Statistics and Machine Learning Toolbox. The MathWorks, Inc., Natick, Massachusetts, United State. URL: https://se.mathworks.com/products/deep-learning.html, Accessed 02.02.20.

[54] Fisher R.A. (1936) The use of multiple measurements in taxonomic problems. Annals of Eugenics 7, pp. 179–188. DOI: 10.1111/j.1469-1809.1936.tb02137.x.

[55] Stehman S.V. (1997) Selecting and interpreting measures of thematic classification accuracy. Remote Sensing of Environment 62, pp. 77–89.

[56] He H. & Ma Y. (2013) Imbalanced learning: foundations, algorithms, and applications. John Wiley & Sons.

[57] Fisher A., Rudin C. & Dominici F. (2019) All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research 20, pp. 1–81.

[58] Oliver D.I. (2014) Privacy engineering: A dataflow and ontological approach. CreateSpace Independent Publishing Platform.

[59] Oliver I. & Miche Y. (2016) On the development of a metric for quality of information content over anonymised data-sets. In: Proc. 2016 IEEE, 10th International Conference on the Quality of Information and Communications Technology (QUATIC), pp. 185–190. DOI: 10.1109/QUATIC.2016.047.

[60] Kraskov A., Stögbauer H. & Grassberger P. (2004) Estimating mutual information. Physical Review E 69, p. 066138. DOI: 10.1103/PhysRevE.69.066138.

[61] Shannon C.E. (1948) A mathematical theory of communication. Bell System Technical Journal 27, pp. 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x.

[62] Pál D., Póczos B. & Szepesvári C. (2010) Estimation of rényi entropy and mutual information based on generalized nearest-neighbor graphs. In: Proc. 2010 NIPS 2010, Advances in Neural Information Processing Systems, pp. 1849–1857.

[63] Finnish diabetes association. URL: https://www.diabetes.fi, Accessed 21.01.2020.

[64] Stoicescu O.M. (2020) Clinflow: An interactive application for processing and exploring clinical data. Master's Thesis, University of Oulu, Department of Computer Science and Engineering .

[65] Jaimes A., Gatica-Perez D., Sebe N. & Huang T.S. (2007) Guest editors' introduction: Human-centered computing–toward a human revolution. Computer 40, pp. 30–34. DOI: 10.1109/MC.2007.169.

[66] Liu X., Liu L., Simske S.J. & Liu J. (2016) Human daily activity recognition for healthcare using wearable and visual sensing data. In: Proc. 2016 IEEE, International Conference on Healthcare Informatics (ICHI), pp. 24–31. DOI: 10.1109/ICHI.2016.100.

[67] Tharwat A. (2018) Classification assessment methods. Applied Computing and Informatics DOI: 10.1016/j.aci.2018.08.003.

[68] Taylor J. (1997) Introduction to Error Analysis, The Study of Uncertainties in Physical Measurements. University Science Book, Mill Valley.

[69] He H. & Garcia E.A. (2009) Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering 21, pp. 1263–1284.

[70] Quintana D. (2019) Synthetic datasets: A non-technical primer for the behavioural sciences to promote reproducibility and hypothesis-generation DOI: 10.31234/osf.io/dmfb3.

[71] Lenert L. & McSwain B.Y. (2020) Balancing health privacy, health information exchange and research in the context of the covid-19 pandemic. Journal of the American Medical Informatics Association .

[72] Ienca M. & Vayena E. (2020) On the responsible use of digital data to tackle the covid-19 pandemic. Nature Medicine 26, pp. 463–464.

[73] United Nations, Department of Economic and Social Affairs, News (2019). COVID-19 – when data save lives. URL: https://www.un.org/development/desa/en/news/statistics/covid-19-when-data-save-lives.html, Accessed 15.05.2020.

[74] Gates B. (2020) Responding to covid-19—a once-in-a-century pandemic? New England Journal of Medicine 382, pp. 1677–1679.

[75] Kucharski A.J., Russell T.W., Diamond C., Liu Y., Edmunds J., Funk S., Eggo R.M., Sun F., Jit M., Munday J.D. et al. (2020) Early dynamics of transmission and control of covid-19: a mathematical modelling study. The Lancet Infectious Diseases .

[76] Wu Z. & McGoogan J.M. (2020) Characteristics of and important lessons from the coronavirus disease 2019 (covid-19) outbreak in china: summary of a report of 72 314 cases from the chinese center for disease control and prevention. The Journal of the American Medical Association 323, pp. 1239–1242.

[77] Baker & McKenzie (2020) COVID-19 Data Privacy & Security Survey. URL: https://www.bakermckenzie.com/-/media/files/insight/publications/2020/04/covid19-data-privacy--security-survey17-april.pdf, Accessed 20.05.20.

[78] Jain S., Moritz D., Halperin D., Howe B. & Lazowska E. (2016) Sqlshare: Results from a multi-year sql-as-a-service experiment. In: Proc. 2016 ACM, International Conference on Management of Data, pp. 281–293.

[79] Bhardwaj A., Bhattacherjee S., Chavan A., Deshpande A., Elmore A.J., Madden S. & Parameswaran A.G. (2014) Datahub: Collaborative data science & dataset version management at scale. arXiv preprint arXiv:1409.0798 .